

YouChoos CT SoundProg

Create your own sounds for CT Elektronik Sound Decoders



Application User Guide

V1.6 3rd July 2013

Copyright © John Gymer, YouChoos, 2011-13

Contents

<i>YouChoos CT SoundProg</i>	<i>1</i>
<i>Application User Guide</i>	<i>1</i>
<i>Contents</i>	<i>2</i>
<i>Introduction</i>	<i>4</i>
DCC – Digital Command Control, for Model Railways	4
CT Elektronik	4
YouChoos	5
CVs – Configuration Variables	6
Bits and Binary	6
Decoders	7
CT SoundProg Device	7
<i>Getting Started</i>	<i>9</i>
Hardware	9
Software	9
<i>Project</i>	<i>11</i>
Opening an existing Project	11
Save a Project	11
Importing CVs and Sound Schedules	11
<i>Load a Decoder</i>	<i>13</i>
Decoder Firmware Updates	13
What is Firmware?	13
How do I get new Firmware versions?	13
Why should I update Firmware?	13
How do I update the Firmware in my Decoder?	13
Loading Projects into a Decoder	15
Preparing for the Load	16
Performing the Load	16
<i>Raw CVs</i>	<i>18</i>
<i>Address</i>	<i>19</i>
<i>Motor</i>	<i>20</i>
Intertia/Momentum	20
Shunting Mode	20
Speed Curves	20
Back EMF and Motor Settings	22
<i>Functions</i>	<i>23</i>
Function Outputs	23
Lighting Effects	23
Dimming Function Outputs	23
Low-Beam Function	24
Automatic Couplers/Uncouplers	24
<i>Sound Schedule & Callable Sounds</i>	<i>25</i>
WAV File Formats	25
Using WAV Files	25

Running Sounds for All Engine Types	26
Steam	27
Diesel/Electric	27
Callable Sounds	28
<i>Sound Configuration</i>	<i>30</i>
Callable Sound Effect ‘Banks’	30
Steam Sound Options	30
Other Options for Sound	31
<i>Function Key Mapping</i>	<i>32</i>
<i>Preferences</i>	<i>33</i>
<i>Version History</i>	<i>34</i>
<i>CV Reference</i>	<i>35</i>
<i>CV.dat File Format</i>	<i>40</i>
<i>Sounds.txt File Format</i>	<i>41</i>
<i>Acknowledgements</i>	<i>47</i>

Introduction

DCC – Digital Command Control, for Model Railways

For many years model railways have been powered and controlled by analogue DC power to the track, with the speed of trains depending upon the voltage supplied to the track. This simple and reliable approach has been employed to great effect on many thousands of railway layouts around the world for decades and is still the preferred choice for many seasoned modellers due to its apparent simplicity.

However, in recent years the modern model railway enthusiast has begun to demand more control, more features and more flexibility. Perfectly reasonable, given how technology has progressed in other walks of life in the meantime. Hence 'DCC' was born (Digital Command Control).

DCC is essentially a messaging system whereby 'instructions' are sent over the track itself, encoded in pulses, and 'decoded' by a device (aptly named a '*decoder*') to understand what is being requested. For example, messages to tell '*Flying Scotsman*' to '*run at half speed*' or '*stop*', or perhaps '*blow a whistle*' or '*switch on lamps*'.

The possibilities of such a messaging system are endless.

Such a technology not only provides the ability to individual control features of a model such as lights, smoke, sounds, coupling devices etc., but also removes the inherent problem with DC layout where only 1 locomotive can be on a 'powered section' of a track at a time. With DCC, it is perfectly feasible to have many locomotives on the same track, but independently controlled. Wiring of power to the track becomes much simpler too – with all sections of the track having power permanently – no more switching of power regions. Essentially, DCC allows you to *drive the train* and not the track!

The NMRA (National Model Railroading Association) worked together with manufacturers such as Lenz to devise a standard for this messaging system, including electrical specifications as well as the messages themselves, and today there are a huge range of different manufacturers producing products that understand how to talk or listen to DCC, including all the major model brands such as Hornby and Bachmann. It is a well-established standard, and is here to stay!

CT Elektronik

CT Elektronik was formed by Cuong Tran (hence 'CT') in Vienna, Austria, designing and manufacturing DCC equipment including decoders and control systems. Tran previously worked at Zimo Elektronik until parting ways to start his own company.



Today, CT produces a range of equipment, including decoders for all scales, which are renowned for their small size and excellent motor control.

All of the current range of decoders follow a unified software version, often referred to as 'Revision 4', although there are physical differences between them – some providing sound, different numbers of function outputs, some with motor control etc.

The YouChoos CT SoundProg application is specifically designed to configure CT decoders that support sound, but many of the other options provided, and techniques used are also applicable to other decoders in the CT portfolio.

YouChoos

YouChoos was formed by John Gymer (that's me!) in Wiltshire, England in 2009, offering bespoke customised models with features such as lighting, smoke generators, sound, detailing, and weathering.



When setting out on the road of deciding which sound decoders to use, I chose to go with CT at the time because the spec of CT sound decoders was on a par with the ESU LokSound and Zimo decoders of the day, but CT decoders were both cheaper, and perhaps more importantly, smaller. The price and the size factors are certainly still true today – size being the aspect that Tran works most rigorously on.

Since 2009 I have worked hard to master the black art of programming sound into these technically great decoders. With limited documentation available, it hasn't been easy.

Hence the reason for creating this application – to make sound project creation on these decoders much simpler and more understandable.

YouChoos now boasts a wide range of UK sound projects for sale with CT decoders, so you can simply purchase a ready-to-plug-in decoder with the sounds of your choice, or if you are feeling adventurous and creative then this application, coupled with the CT SoundProg device opens up the possibility of building and loading sounds yourself.

CVs – Configuration Variables

The 'CVs' of a DCC decoder are essentially a collection of parameters/variables which influence how the decoder behaves. Many CVs are standard, in that they follow the guidelines set out by the NMRA. However, any fairly advanced decoder will have in addition to this a whole bunch of CVs specific to itself. This is especially true for sound decoders, which provide far more functionality than was ever designed for when NMRA published the original standards.

Each CV is referred to by its *index*, or *number*, which gives it its' own unique meaning. For example, *CV number 1* is the decoder's *short address*, and *CV number 3* defines the *rate of acceleration* (used to simulate a momentum effect), and so on. A full list of CVs supported in the CT sound decoders is supplied in a reference chapter towards the end of this document.

The purpose of this application is to present these CVs in a more understandable format, with readable descriptions and a simple interface to modify them.

Bits and Binary

In order to successfully understand and program some CVs, it would be beneficial to have a basic understanding of *binary*. Each CV contains what is called a *byte* of information. This is computer-speak for 8 *bits* of information, each of which can be *ON* or *OFF*. A *bit* is therefore like a *light switch*, ON or OFF. A *1* represents ON and a *0* represents OFF. If you have just 1 bit, then you can have a maximum of 2 values i.e. on and off. Adding more bits means you can have more combinations, for example, 2 bits gives you 4 possible combinations: OFF+OFF; OFF+ON; ON+OFF; ON+ON, or 0,0; 0,1; 1,0; 1,1. Read this as 0,1,2,3 since computers always start at 0 instead of 1.



By convention, bits are read with the least significant to the right i.e. "bit 0" is the right-most bit. A *byte*, as mentioned previously has 8 bits, or bits 0 to 7, giving a possible range of 0-255 ($(2^8) - 1$ being the maximum value i.e. 256 combinations). Use the table below for reference to see what value each bit can represent.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
128	64	32	16	8	4	2	1

An example: if bit 6 is ON and bit 0 is also ON, then this is $64+1$, so the value represented is 65. Simple really! If all 8 bits are switched on, adding them up gives 255. With no bits switched on, you have a value of 0.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
128	64 ON	32	16	8	4	2	1 ON

Many of the CVs in your decoder use individual bits to control different aspects, so it is useful to understand binary in order to a) work out how the decoder is currently configured, and b) to understand how to modify the CVs to change the decoder's behaviour.

This application will of course hide much of this from you, but even so, it is valuable to at least have some understanding.

Decoders

Revision 4 CT sound decoders are the intended target for this application, which includes the following specific models:

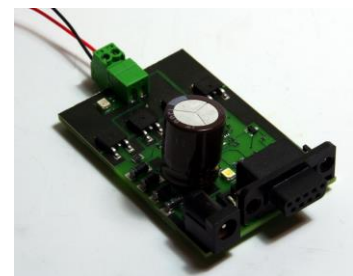
- SL51-4 – an 8-function motor & sound decoder ideal for use in OO/HO scales
- SL75 – a 4-function micro motor & sound decoder ideal for use in N scale or small OO models
- GE75 – a 2-function version of the SL75, which has sound features, but no motor control – ideal in N or OO scales as a sound-only device
- SL76 – a 4-function sub-miniature motor & sounds decoder ideal for use in N scale or even smaller (is a replacement for the SL75 decoder)
- GE76 - a 4-function sound (but no motor) version of the SL76
- SL82 – a 9-function large-scale decoder which comes in 2.5A or 4A version ideal for O gauge



Note that this application does NOT support older decoders such as SL51-2 or GE70. It also has a requirement that the firmware of the decoder is at version 40 or later (although firmware updates are possible through the application anyway if you have a decoder with an older firmware). More is discussed on the subject of Firmware in the *Load* chapter later on.

CT SoundProg Device

CT's *SoundProg* is a small electronic device that is plugged into your PC, and also to your decoder, and is required to load sounds into the decoder. There is no other device that allows you to do this, so don't be tempted to try using Zimo's equivalent device, or ESU's LokProgrammer – they are entirely proprietary and mixing devices with decoders for other manufacturers may destroy the decoder or the device, or both!!



SoundProg connects to the PC via a Serial Cable, which is now rather ancient technology, and many PCs don't provide a Serial port. Fear not though, a special cable that convert Serial to USB is available so you can simply plug your SoundProg into an available USB port on the PC. Beware though that not all Serial-to-USB cables will work – they require a particular mapping of wires/pins. YouChoos recommends any Serial-to-USB cable that uses the Prolific 2303 chip-set, but others may also be suitable. If you purchased your *SoundProg* from YouChoos then this will probably have been supplied in your pack.

SoundProg also requires a suitable power supply to be connected. A 7.5W 500mA AC (15v) supply is the ideal rating, although apparently an 18V DC supply will also work. Again, if you purchased your *SoundProg* from YouChoos then you'll likely already have this in your pack.

The other side of the *SoundProg* device is where the decoder is connected, ready for programming. 2 wires are needed – left track and right track. In fact, you can program a decoder just as easily if it is already installed in a loco simply by connecting these 2 wires to you track instead of directly to the red + black wires of the decoder.

The decoder must, however, have a speaker attached during programming, or you won't know that it has correctly been updated – a series of audible beeps can be heard during the programming cycle that indicate progress, success or failure.

SoundProg itself is a simple device, and provides no feedback to the PC whatsoever to indicate how it is getting on – this is all done via the audible beeps. More on that later (in the '*Load a Decoder*' chapter).

Getting Started

Hardware

Before plugging in the Serial-to-USB cable to your PC, you should first install the *Prolific 2303* Device Driver for that cable. This should have been supplied on CD, or binary distribution with your package. Use the appropriate Device Driver for your version of Windows.

Once installed, you can connect the Serial-to-USB cable to any available USB port in your computer.

Next, connect the Serial connector of this cable to the CT SoundProg device.

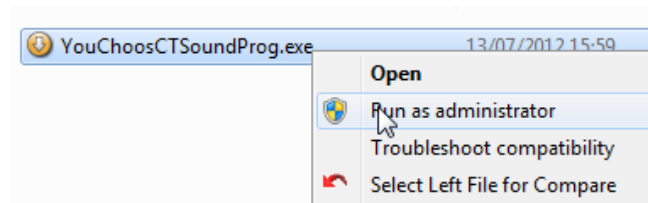
Note that you only need to connect the power cable into the CT SoundProg device when you want to perform any loading into a decoder. It is recommended that you leave the power cable disconnected at all other times. This is because the CT SoundProg device is an open-circuit device and this practise will help avoid possible short-circuits by mistake!

When powered correctly, the device should have a single yellow (or white in some versions of the device) LED illuminated near the decoder output end.

Software

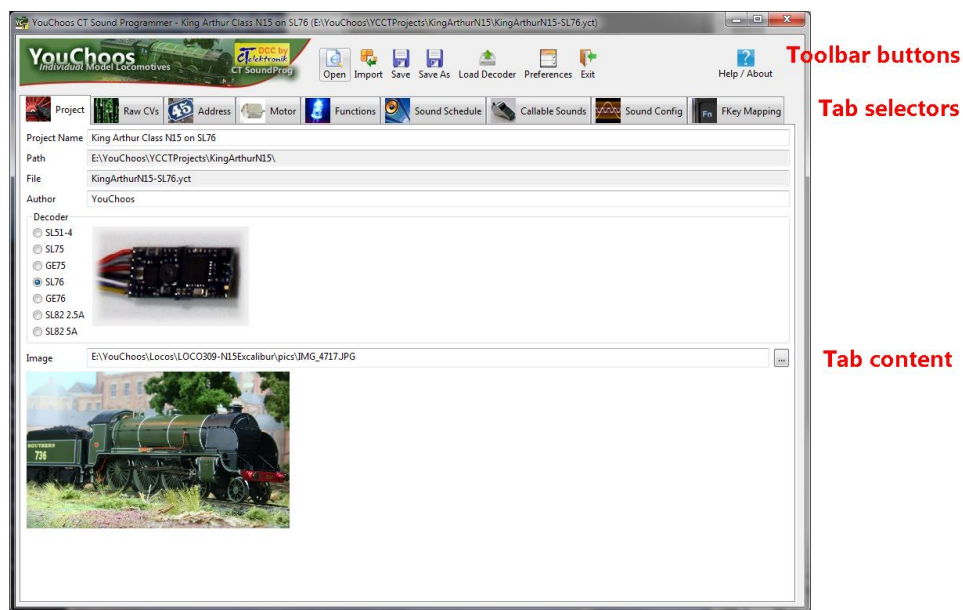
Run the YouChoos CT SoundProg application installer supplied either on CD with your package, or via binary distribution (probably e-mail). Follow the prompts and choose where you would like the software installed (default will be under your *Program Files* directory).

For Vista or Windows 7 you may need to run the installer *As Administrator*.



After installing the YouChoos CT SoundProg application, a shortcut icon will be created in your Windows Start menu. Choose this icon to launch the application.

The layout is very simple, as is shown in the figure below:



Along the top are some toolbar icons, which perform high-level functions, such as open/save Project etc. The main portion of the window is where you configure the details of the Project, and is presented as multiple *tabs*. Clicking each tab switches to the details relating to that topic.

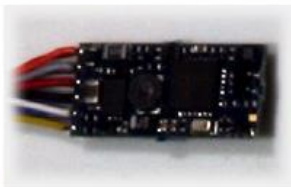


Project

A *Project* is defined within a file which has an extension of “.YCT”. This definition file lists the CV values for that Project, along with a list of the sound sample files that it uses, as well as some high-level information about the Project.

A Project can also have other high-level information stored against it, such as a Name, Author and a picture to help you identify what type of locomotive the Project relates to.

In order to save/store the Project you must also give it a File, which has a YCT extension.

Project Name	King Arthur Class N15 on SL76
Path	E:\YouChoos\YCCTProjects\KingArthurN15\
File	KingArthurN15-SL76.yct
Author	YouChoos
Decoder	<div> <input type="radio"/> SL51-4 <input type="radio"/> SL75 <input type="radio"/> GE75 <input checked="" type="radio"/> SL76 <input type="radio"/> GE76 <input type="radio"/> SL82 2.5A <input type="radio"/> SL82 5A </div> 
Image	E:\YouChoos\Locos\LOCO309-N15Excalibur\pics\IMG_4717.JPG 
	

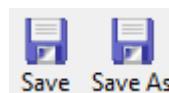
Opening an existing Project

Use the OPEN button at the top of the window to locate and open an existing Project. Any values currently defined will be lost unless you save them as another Project file first!



Save a Project

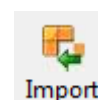
If you loaded an existing Project to start with, you can simply press the SAVE button at the top of the window to save your changes to the same Project. Otherwise you can use the SAVE AS button to create a new Project file.

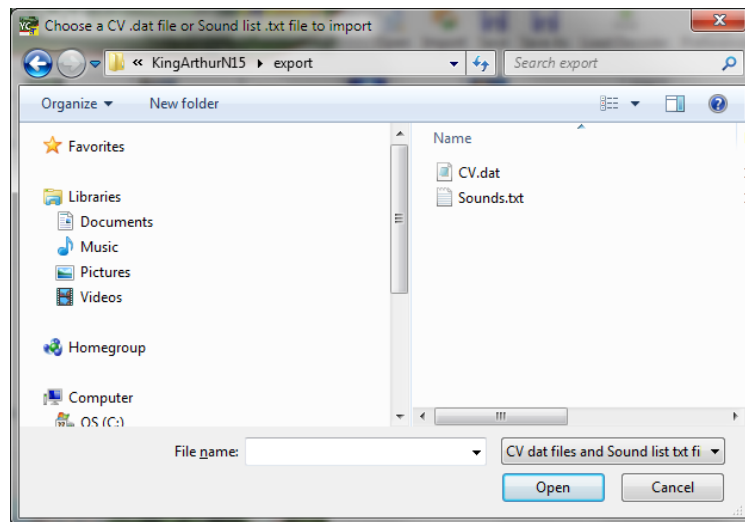


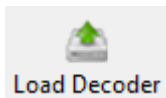
You can save your YCT Project files anywhere on your file system, but it probably makes most sense to save them all in the same directory, or perhaps within a directory structure that makes sense to you.

Importing CVs and Sound Schedules

Prior to this application being available, the most common way to develop Projects for CT decoders was to edit 2 text files by hand – one that listed the CV values and another that listed the sound samples used. If you have any such old Project files then you can import the CV.dat file or the SOUNDS.txt files independently into your Project.







Load a Decoder

Decoder Firmware Updates

What is Firmware?

The *Firmware* of a decoder is the internal software stored inside the decoder that processes DCC messages, translating them into real actions, such as motor control, function outputs and sound etc. The decoder is essentially a tiny computer, just like your PC, and the firmware is its *Operating System* (like Windows, but much smaller, and doesn't crash so often!!).

How do I get new Firmware versions?

From time to time, CT will release new versions of firmware for each decoder, introducing bug fixes and enhancements, which you may wish to have.

Each time YouChoos is informed of new firmware versions, a new version of this application will be made available for download and you can simply reinstall it over your existing copy (no settings will be lost). You do not need to uninstall the application.

Appropriate versions of firmware are listed for each type of decoder in the LOAD DECODER dialog.

Why should I update Firmware?

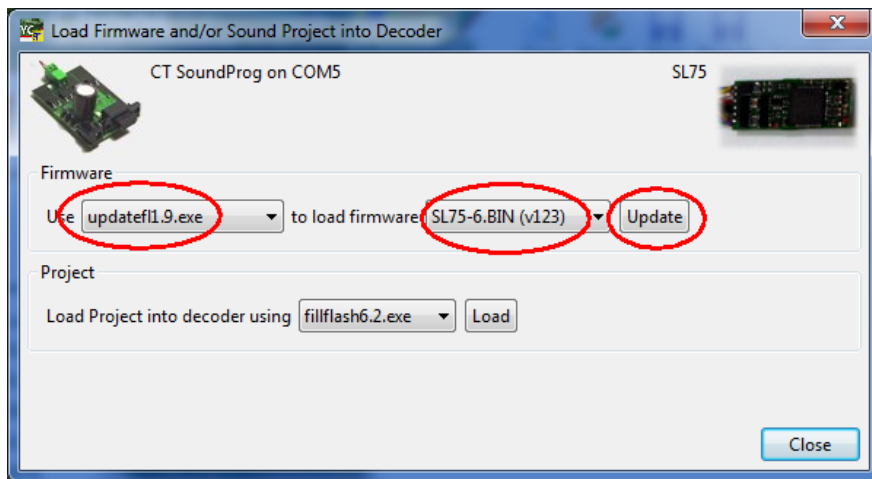
In most cases it will not be necessary to update the firmware, as new decoders are generally supplied with the latest version at the time anyway, but in some cases you may wish to update. For example, many of the early revision SL76 decoders were shipped with V117 firmware, but need to be updated to at least V123 in order for loading of sounds to work at all! That's an example of a bug fixed in the firmware. It won't do any harm to update the firmware to the latest before loading your sound Project anyway, and is good practise, although not always necessary.

In addition, if you use a Windows 7 or 64-bit edition of Windows, you will need to use at least V127 of the firmware, or you will likely encounter problems loading your projects.

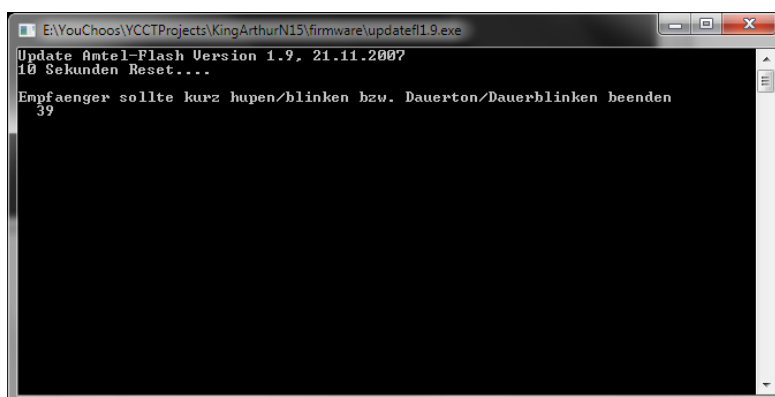
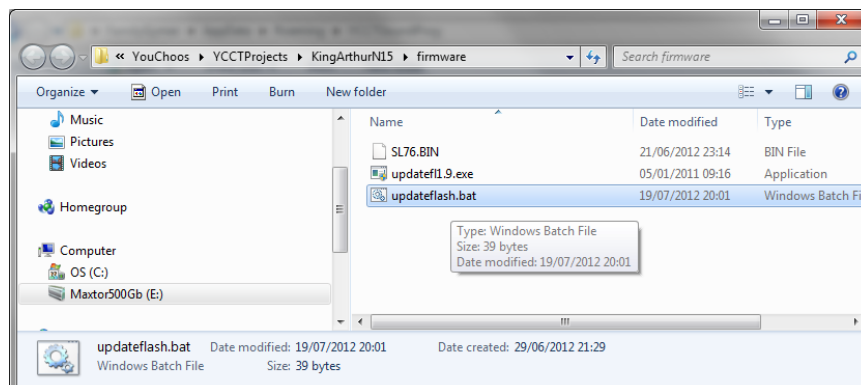
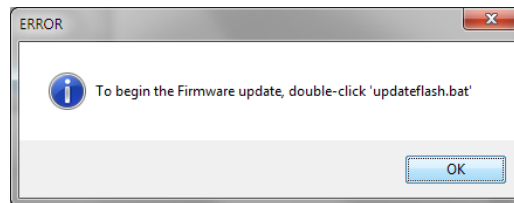
How do I update the Firmware in my Decoder?

To update a decoder's Firmware, use the LOAD DECODER toolbar button at the top of the window. The dialog is divided into 2 parts – one for updating the Firmware, and another for load the sound Project itself.

To update the Firmware, simply choose which version of the Firmware you want to use (probably the latest!) and press the UPDATE button. You also have the option of choosing a version of the CT program to use to actually perform the update. You should not need to change this, and the latest appropriate version for your decoder should already be automatically selected for you.



On pressing the UPDATE button, a subdirectory called '*firmware*' will be created, beneath the location of the Project's YCT file, and all necessary files will be created there, ready for you to run. Windows Explorer will open automatically at the directory and you can simply run the script "updateflash.bat".



It takes 30-60 seconds to update a decoder's Firmware, and progress is indicated in a black box that appears on the screen (a Command Prompt). The messages are in German, but you can probably guess what most of them mean!

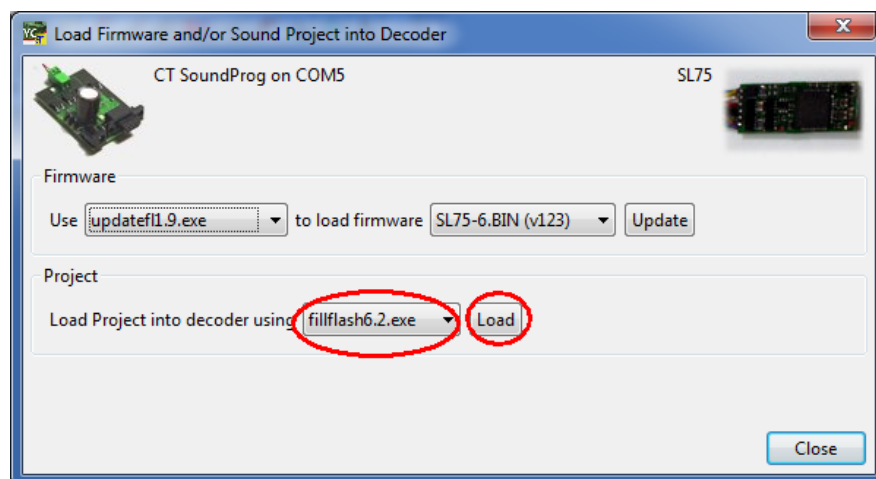
At the start of the update cycle the SoundProg device will switch off all of its LEDs for 10 seconds (this is a reset period), and then the actual update of the Firmware begins, usually indicated by a single beep sound from the speaker attached to your decoder (you should always have a speaker attached to the decoder during the Firmware update process). If you do not hear this single beep at all, it may be because the decoder wires are connected the wrong way around – try swapping the red and black wires around and try again – the SoundProg device is sensitive to this, and programming will only work with it connected a particular way around!

At the end of the cycle, a message will appear stating that it has finished ("Fertig" in German), and the decoder will normally sound a very long beep (10-20 seconds long). When it eventually goes quiet, it is a good idea to take the power out of the SoundProg device before reinserting it ready for the sound Project loading. Note that when updating an SL51-4 decoder to V127 firmware, the 'finished' beep will continue to sound until you take away the power – so just unplug the power lead from the SoundProg device, and reinsert again.

If anything goes wrong during the Firmware update, you will either receive messages on the PC, or you will not get the long beep at the end of the cycle. If the decoder detects a definitive error and cannot be updated, you will often get a low-pitch beep, or series of beeps. Check that your connected decoder matches the type shown on the screen.

Loading Projects into a Decoder

To load a sound Project into your decoder, ensure that it is all connected and that the power supply is inserted into the SoundProg device, then use the LOAD DECODER toolbar icon at the top of the window.



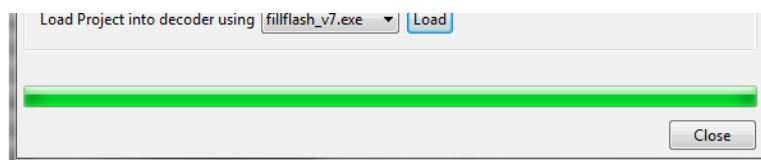
Various versions of the program that performs the load are available from CT, but the latest one appropriate to your decoder should be automatically selected for you, so you shouldn't need to change it. Most decoders at the time of writing use a version called *fillflash_v7*, although this assumes that the firmware of the decoder is first updated to V127 or later.

Preparing for the Load

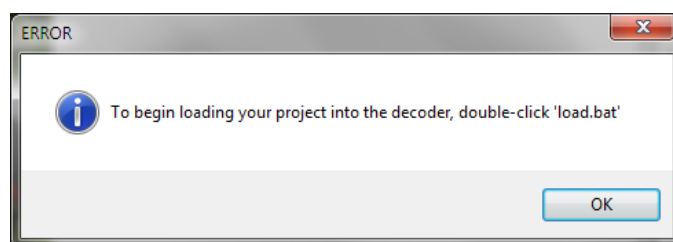
Simply press the LOAD button to prepare everything ready for loading.

A progress bar will move along the bottom of the window to show how far this preparation process goes. Behind the scenes it is creating a subdirectory called '*export*' in the location where your Project's YCT file is stored. Into this directory will be placed:

- the CV.dat file that defines the values of all CVs in your Project
- a Sounds.txt file that lists the sound sample files required, and how that are used in the Project
- the chosen FillFlash executable which actually loads the Project
- a batch script file called *load.bat* which is what you need to run to start the load process
- copies of all of the sound sample files (WAVs) required by the Project

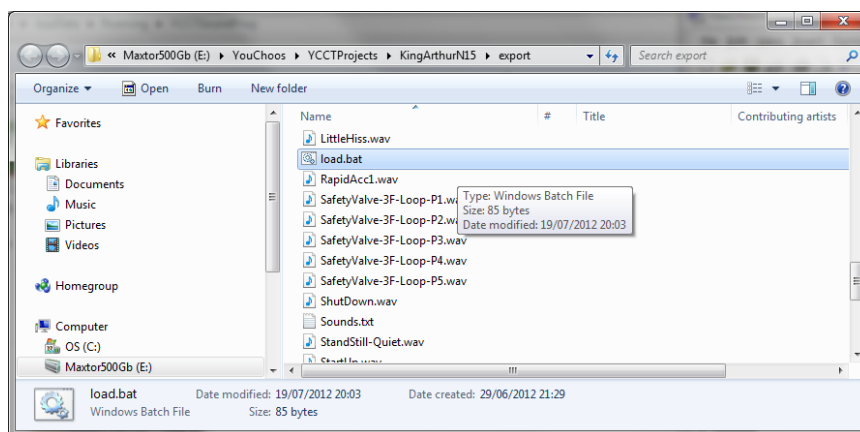


Note that any existing content of the *export* folder will be erased when you hit the LOAD button!



Performing the Load

Once the preparation is complete, you need to run the *load.bat* script file located in the *export* directory. This starts CT's *FillFlash* executable to actually communicate with the decoder, and will appear in a black box window (Command Prompt), which shows progress, or any errors.



All FillFlash output is in German, but don't worry, in general you don't need to understand what it is saying!

At the start of the programming cycle, all LEDs on the SoundProg device will switch off for 10 seconds, which is a reset period to make sure everything is initialised before it starting sending data to the decoder.

If your decoder has the analogue mode switched on (CV29 bit2 ON), then you probably won't be able to program sounds until you switch analogue model off. The loco will most likely shoot off uncontrollably as soon as you put it on the track! This is due to the decoder trying to check whether it is running on DC or DCC at startup, and the SoundProg output is different from either of these, so it gets confused!

After this period, a single beep will be made from the decoder to indicate that it is happy to start the programming cycle. If you do not hear this single beep at all, it may be because the decoder wires are connected the wrong way around – try swapping the red and black wires around and try again – the SoundProg device is sensitive to this, and programming will only work with it connected a particular way around, and with a speaker attached!

Lots of output will then appear and scroll to indicate progress. If there is a problem with your Project definition then the process will normally fail very early on, and there are lots of reasons that this may happen. Here are some of the common reasons:

- Total WAV sample size too large
- WAV file not found
- Sound sample format incorrect (needs to be 8 or 16-bit mono, and 22kHz or 11khz)
- Invalid value for a CV

If you see these, go back and fix the Project and try again!

Once the programming cycle begins, you'll see one or more of the SoundProg device's LEDs flashing or flickering slightly.

Depending upon the size of the Project, it could take anything from 60 seconds up to 20 minutes to perform the whole load cycle, but at the end of the cycle the decoder should beep frantically a

number of times at the same pitch as the original starting beep. If you get a lower-pitched beep then this indicates an error.

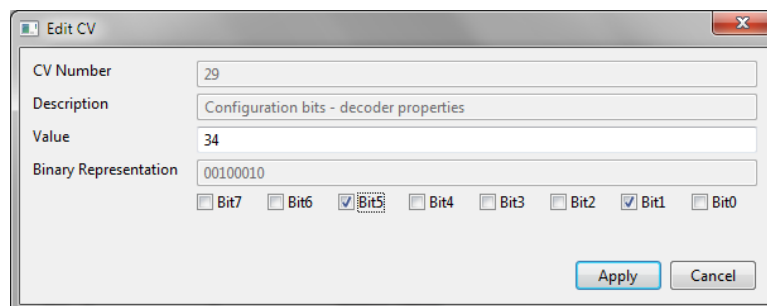
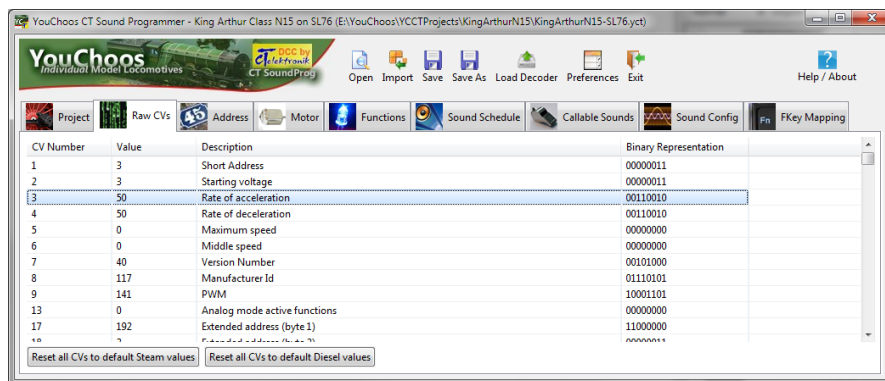
Note that SL76/GE76 gives a lower-pitched beep than SL51-4, SL75 or GE75.

Also note that SL82 does not beep at all at the start of the cycle.



Raw CVs

The *Raw CVs* tab presents all CVs in the Project in a simple form, allowing you to edit each CV individually. By double-clicking a CV entry, it will pop-up a dialog box where you can enter a value for the CV as numeric, or as binary.



Note that for most CVs, there are more convenient ways of setting their values in other tabs.

CVs may have any value from 0 to 255 assigned, but the actual meaning of the value will vary, and certain ranges may not be valid for particular CVs. For example, CV 121 controls the volume of the main running sounds, which takes a value in the range 0 to 63, so anything above 63 is not valid for that CV. Check the decoder manual for details, or the appendix at the end of this user guide.



Address

In general, each decoder on your railway layout will be assigned a different DCC address, which allows your digital controller to send individual messages to specific decoders.

Short or Long Address			
<input checked="" type="radio"/> Use Short Address	<input type="text" value="3"/>	CV3	<input type="text" value="3"/>
<input type="radio"/> Use Long Address	<input type="text" value="3"/>	CV17/18	<input type="text" value="192/3"/>

All decoders ship with a default address of 3, so you will probably need to change your decoder's address. 2 numbering schemes are available: *short* addressing (1-127) and *long* addressing (128-9999). Actually the ranges extend slightly beyond this, but you should try to keep with those address ranges.

Depending upon your digital controller's capabilities you will choose what addresses to use. For example, Bachmann Ez-Command can only address decoders in the range 1-9. Other controllers can only access 2-digit addresses 1-99. Most modern controllers will allow addressing the entire range 1-9999 though, and it is convenient to use part of the cab number of the loco as the address, as it is easy to remember.

In the *Address* tab, choose whether you want to use a *Short* address, or *Long* address. Both are stored, and you choose which one is active with Bit5 (value 32) in CV29 – you don't have to remember this – it is done automatically by choosing the desired radio button!

In case you are interested, the *Short* address is stored in CV1, and the *Long* address is stored in CVs 17 and 18 (it requires 2 CVs to get enough combinations of bits... remember how a single CV is made up of 8 bits giving 256 different combinations? Using 2 CVs means 16 bits, so a potential range of 65535 combinations is possible in fact).



Motor

Inertia/Momentum

Inertia is the effect created where an object has kinetic energy (movement) and mass – a heavy object is harder to get moving (or to slow down) than a lighter object. In traditional DC model railways you could turn up the power very quickly and the loco would lurch unrealistically off at full speed! DCC provides a built-in feature to simulate the effect of inertia (or momentum) causing a loco to gradually build up to the requested speed, and to gradually reduce its speed when requested to slow down.

CT decoders offer this support and you can vary the effect of inertia independently for acceleration vs deceleration – it might be a heavy diesel with fantastic breaks for example!

Inertia/Momentum
Inertia effect during acceleration (CV3)
Inertia effect during deceleration (CV4)

Inertia is stored in CVs 3 (acceleration) and 4 (deceleration). A value of 0 means no effect, so it will behave similar to DC analog control when you give it some welly!

Shunting Mode

If you have assigned one of the Function Keys to toggle *Shunting Mode* on/off, then you can control what effect this has using the *Shunting Mode* options.

Shunting Mode Options
☐ Inertia disabled while in shunting mode (CV116 Bit0)
☐ Max speed halved in shunting mode (CV116 Bit1)
☐ Reverse is 65% of forward speed (CV116 Bit2 - regardless of shunting mode)

CV116's Bit0 and Bit1 control these effects. If neither of these Bits is set then there will be no change in behaviour at all when in shunting mode.

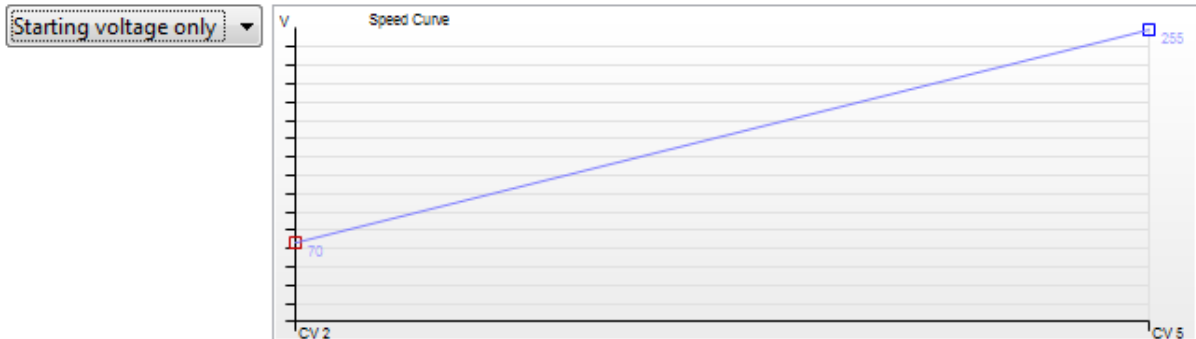
The 3rd checkbox (*reverse is 65% forward speed*) is actually applied regardless of whether you are in shunting mode or not – but it is related, so it is placed together with the shunting mode options.

Speed Curves

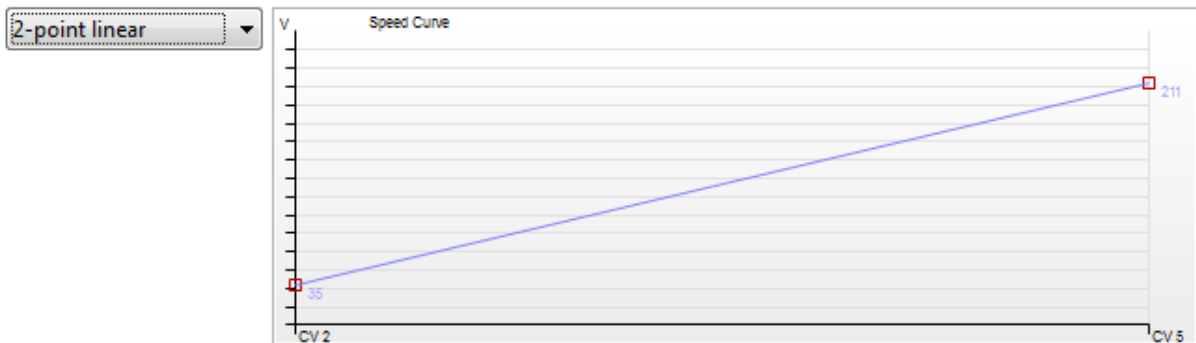
By default, the decoder will use a linear algorithm when converting request speed step to motor current, with speed step 0 giving 0 current, and speed step 127 (top) giving 100% motor current.

For many models this will work just fine, but in some situations you may want to tweak this curve, and there are a number of ways in which it can be adjusted...

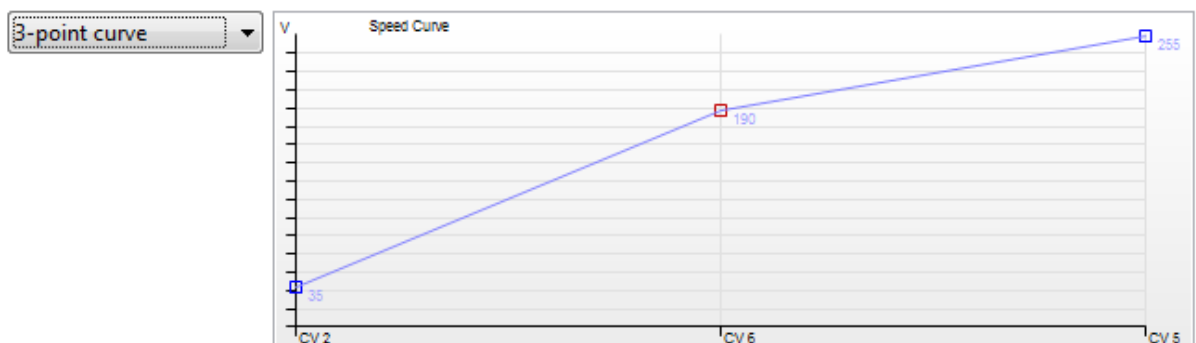
- Starting Voltage only – by choosing this option, top speed will always be 100% motor current, but the starting speed (speed step 1) will be at a configurable level, with everything in between being linear between those 2 points. This is particularly useful for models that require a bit of grunt to get them to physically start moving.



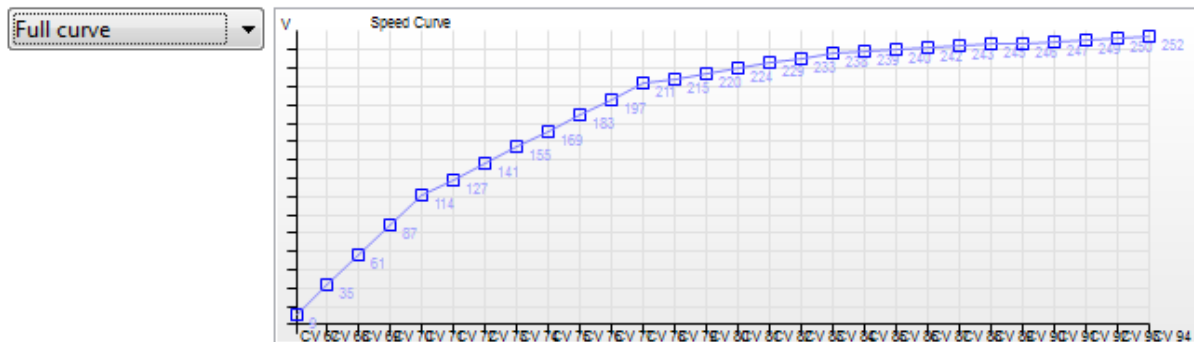
- 2-point linear – this gives a similar linear control of motor current output, but allows you to specify a starting voltage and a max voltage, effectively limiting top speed.



- 3-point curve – where a motor's characteristics are not linear, it may be enough to use a 3-point curve where you specify a start voltage, mid-point voltage and max voltage. This gives 2 linear lines below and above the middle speed step.



- Full curve – use this where you want total control of the speed step vs motor current output throughout the range. As you click individual points, the curve is automatically adjusted around those manually chosen point, making it easier to draw a good shape. Going to this extreme is not normally required however, and for most situations, a 3-point curve is probably sufficient.



Back EMF and Motor Settings

Back EMF is a technique used by DCC decoders to measure the actual current being drawn by a motor – effectively a way to measure how hard it is working. It can then be used to automatically adjust the motor output, the intention being to keep the real speed of the loco at the requested speed – regardless of load, hills, curves, or other factors.

Back EMF is measured by sending tiny pulses to the motor connections and measuring the response, so it cannot be done constantly. Therefore it sends regular pulses for the measuring, but not so regular as to disrupt the flow of current to the motor. The *Motor PWM* setting in CV9 controls the frequency at which this occurs. Most modern motors work well in a range of 134-160, but you may need to adjust this to get better results out of other motors, in particular bell, anchor or coreless motors.

EMF and other Characteristics		
Motor PWM	<input type="text" value="141"/>	(CV9 13-63 for stepless 30-150Hz and 141-191 for coreless/anchor 16kHz)
Back EMF Intensity	<input type="text" value="255"/>	(CV50)
EMF P-Value (Proportional part)	<input type="text" value="100"/>	(CV51)
EMF I-Value (Integral part)	<input type="text" value="0"/>	(CV52)
Reference Voltage	<input type="text" value="90"/>	(CV64)
<input type="checkbox"/> Reverse direction (CV29 Bit0)		
Speed steps (CV29 Bit1)	<input type="text" value="28/128 steps"/>	
Motor control frequency (CV137 Bit7)	<input type="text" value="16kHz"/>	

After measuring the EMF, its value is used to affect the motor output, and the level to which this is applied (fed-back) is controlled in CV50. CVs 51 and 52 can also be tweaked to alter the frequency at which the sampling is performed. Typical values for the most common motors are CV51=80 and CV52=40, but you can experiment with these values to see the effect, and to tune it better if necessary.

Other motor-related features are available in this tab too – see the CV reference for further information.



Functions

Function Outputs

The *Functions* tab is where you define what the various function outputs of the decoder are used for. Depending upon the type of decoder you might have anything from 2 up to 9 function outputs available. This application provides facilities to manage up to 8 outputs.

Function outputs can be used for lights, smoke generators, automatic coupling devices, or any manner of other physical/electrical devices.

The top part of this tab allows you to document what each function output does in this specific decoder. By convention, the first two outputs are used for directional lighting, which correspond to the WHITE and the YELLOW output wires.

Output	Feature Description	Lighting Effect	Direction Active	Apply Dimming	LowBeam	Coupler Dim	Coupler FWD	Coupler REV
Aux0 Fwd White	Forward motion lights	Constant light	Forward	<input type="checkbox"/> Dimmed	<input type="checkbox"/> LowBeam	<input type="checkbox"/> Dim	<input type="checkbox"/> FWD	<input type="checkbox"/> REV
Aux1 Rev Yellow	Reverse motion lights	Constant light	Reverse	<input type="checkbox"/> Dimmed	<input type="checkbox"/> LowBeam	<input type="checkbox"/> Dim	<input type="checkbox"/> FWD	<input type="checkbox"/> REV
Aux2 Green	Firebox glow	Gyalite	Both	<input type="checkbox"/> Dimmed	<input type="checkbox"/> LowBeam	<input type="checkbox"/> Dim	<input type="checkbox"/> FWD	<input type="checkbox"/> REV
Aux3 Purple	<undefined>	Constant light	Both	<input type="checkbox"/> Dimmed	<input type="checkbox"/> LowBeam	<input type="checkbox"/> Dim	<input checked="" type="checkbox"/> FWD	<input checked="" type="checkbox"/> REV
Aux4 Pad 1	<undefined>	Constant light	Both	<input type="checkbox"/> Dimmed	<input type="checkbox"/> LowBeam	<input type="checkbox"/> Dim	<input type="checkbox"/> FWD	<input type="checkbox"/> REV
Aux5 Pad 2	<undefined>	Constant light	Both	<input type="checkbox"/> Dimmed	<input type="checkbox"/> LowBeam	<input type="checkbox"/> Dim	<input type="checkbox"/> FWD	<input type="checkbox"/> REV
Aux6 Pad 3	<undefined>	Constant light	Both	<input type="checkbox"/> Dimmed	<input type="checkbox"/> LowBeam	<input type="checkbox"/> Dim	<input type="checkbox"/> FWD	<input type="checkbox"/> REV
Aux7 Pad 4	<undefined>	Constant light	Both	<input type="checkbox"/> Dimmed	<input type="checkbox"/> LowBeam	<input type="checkbox"/> Dim	<input type="checkbox"/> FWD	<input type="checkbox"/> REV

Lighting Effects

Use the *Lighting Effect* drop-downs to define any special lighting effects you wish to apply to each function output. Numerous flashing and glowing effects are possible.

For any outputs that you choose to apply a flashing effect to, these can be tuned by setting a *Flash Low Point* (CV114) and a *Flash Cycle* time (CV115).

Flashing Properties (affects all outputs with flashing effects)

Flash Low Point % (CV114)

Flash Cycle ms (CV115)

The *Direction* drop-downs allow you to choose which direction each of the outputs should be active for.

Dimming Function Outputs

On occasions it may be necessary to modify a function output's maximum level, such as dimming a bright light, or reducing the amount of current to a smoke generator. This is achieved by applying *dimming* to one or more function outputs. Choose which outputs have this applied with the *Dimmed* checkboxes, that define the level to which you'd like them dimmed using *Dimming Level* (CV54).

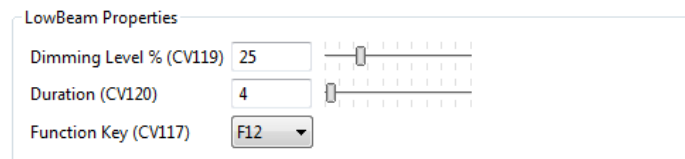
Dimming Properties (affects all outputs with dimming applied)

Dimming Level % (CV54)

Note that this *Dimming Level* will be applied to ALL function outputs that have the *Dimmed* checkbox switched on.

Low-Beam Function

In addition to permanent dimming as described above, it is possible to define one or more of the function outputs to have temporary dimming applied by pressing a FunctionKey. For example, you could dip the headlight temporarily by pressing a FunctionKey, and it would reduce by the level defined in *Dimming Level* (CV119) for the *Duration* defined in CV120. The FunctionKey is selected with CV117, where a value of 0 means that no FunctionKey is assigned for this purpose.



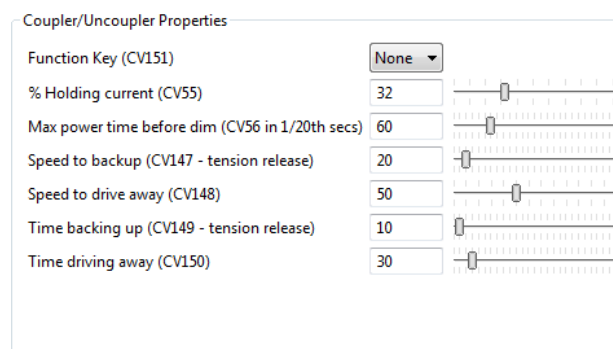
LowBeam Properties	
Dimming Level % (CV119)	25
Duration (CV120)	4
Function Key (CV117)	F12

Choose which function outputs have this effect applied by using the *LowBeam* checkboxes.

Note that not all types of decoders appear to implement this feature.

Automatic Couplers/Uncouplers

CT decoders have some great support for automatic coupling devices too. Use the checkboxes *Coupler Dim*, *Coupler FWD* and *Coupler REV* to define which function outputs are used for coupling devices, then the various CVs in the *Coupler/Uncoupler Properties* group to decide how they actually work.



Coupler/Uncoupler Properties	
Function Key (CV151)	None
% Holding current (CV55)	32
Max power time before dim (CV56 in 1/20th secs)	60
Speed to backup (CV147 - tension release)	20
Speed to drive away (CV148)	50
Time backing up (CV149 - tension release)	10
Time driving away (CV150)	30

You may have automatic couplers at each end of your loco, so the current direction of travel is important in which output gets activated when you hit the FunctionKey defined in CV151.

When you do this, the loco will first drive towards the coupled wagon/coach (speed and duration defined in CV146 and CV149), which has the effect of releasing the coupling's tension, then the function output will be triggered at full power for a short period (as defined in CV56), which hopefully will move the coupling's magnet high enough to clear the connection. After this the loco will automatically drive away from the wagon/coach (how fast and long is defined in CV148 and CV150).

It is a very neat effect, and when tuned carefully get result in very clean decoupling indeed!



Sound Schedule & Callable Sounds

The heart of any sound Project is obviously the sounds themselves! This application presents the sounds in 2 separate categories... the main running sounds, and the callable sound effects.

Depending upon what type of engine you are recreating, you'll get a different *schedule* for steam vs diesel, but the principles are similar, and the interface in how you select sounds into the schedule is the same.

Sounds are further categorised into specific features, such as *hard braking*, or *start-up sequence* etc. These specific features are presented in coloured boxes, laid out in an easy-to-understand format.

Sounds related to acceleration are in green, deceleration in red, constant speed sounds are in grey and callable sounds are in blue.

Each sound may consist of anything from a single WAV file up to 5 WAV files, which make up the whole sound effect, and for sounds comprising of 3 or 5 individual WAV files, these can often be *looping* sounds, where the middle sound is repeated a number of times before the overall effect continues to the remaining WAV files.

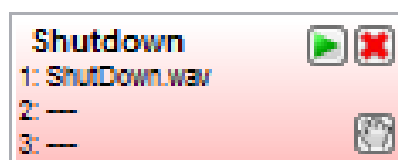
WAV File Formats

WAV files may be 16-bit or 8-bit mono sounds, in either 22KHz or 11KHz frequencies. Use 22KHz where possible to preserve maximum quality and definition, but use 11kHz where remaining space is low and the sound is not too adversely affected by reducing the frequency. Note that all sounds are internally converted to 8-bit at load time anyway.

Using WAV Files

To place a WAV file into a particular *slot*, use the File Explorer on the left panel of the window to browse your local files, then click and drag the WAV file into the slot of your choice. An empty slot is depicted by "---".

The meaning of each sound is described with a title at the top of the box, for example *Shutdown* in the example below:



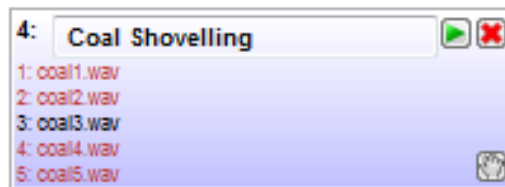
In this example we have chosen to fill just the 1st slot in the box with a WAV file called *ShutDown.wav*.

Use the PLAY button to listen to the sounds through the PC audio.

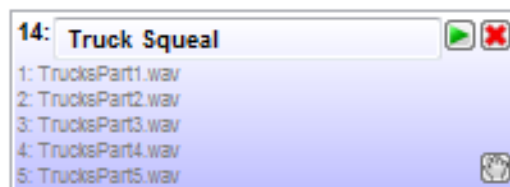
Use the Red X button to CLEAR all WAVs from the box, emptying it totally.

It is easy to move WAV files around different slots in the same box, or even to slots in other boxes simply by dragging the individual slots. You can even drag an entire box's content to another box by dragging from the little hand icon in the bottom right of the box.

Where a file is not in a usable format, such as being a 44kHz WAV instead of the supported frequencies of 22kHz or 11kHz, it will be displayed in RED as below. You will not be able to perform the load of the Project if any WAV files are of an unsupported format.



You can also easily see which WAVs are 22kHz vs 11kHz since the 11kHz WAVs are displayed in grey instead of the usual black text:



Clicking any WAV file will cause a miniature graphic of the sound to appear in the top at the centre of the display, with each of the constituent WAV files in alternative RED/GREEN colours. The example below shows 4 chuff beats:



Running Sounds for All Engine Types

Many of the sound *boxes* are common to all types of engines, diesel, 2-cylinder steam, 3-cylinder steam or 4-cylinder steam. These are as follows:

- STARTUP
- STANDSTILL/IDLE
- CHANGE DIRECTION
- SET OFF (channel 1)
- SET OFF (channel 2)
- RAPID ACCELERATION
- RAPID DECELERATION
- COME TO HALT (channel 1)

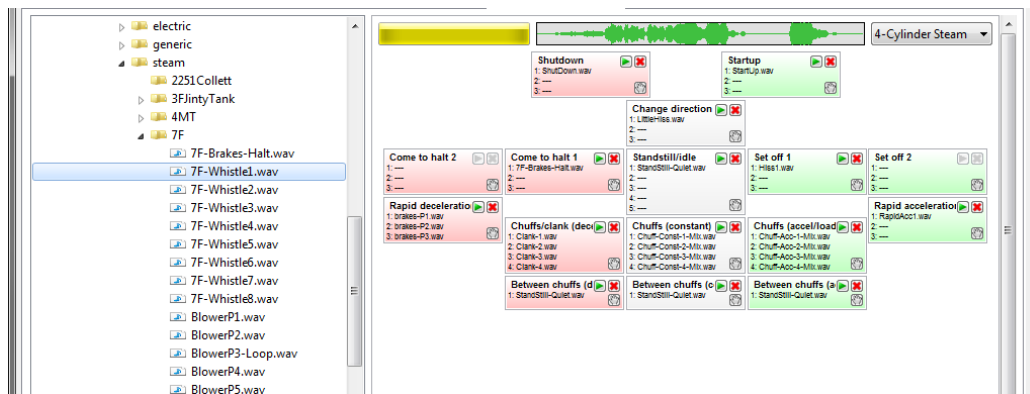
- COME TO HALT (channel 2)
- SHUTDOWN

Steam

For steam Projects, you can choose between 2-cylinder, 3-cylinder or 4-cylinder. In fact, it is often quite effective to choose 4-cylinder even for 2-cylinder trains, as the real thing has 4 different chuff tones as the wheels rotate (each cylinder gets 2 chuffs per wheel revolution).

The drop-down in the top-right of the schedule defines which type of engine you are creating.

You must fill in ALL slots for the chuffs according to the type of steam engine selected.



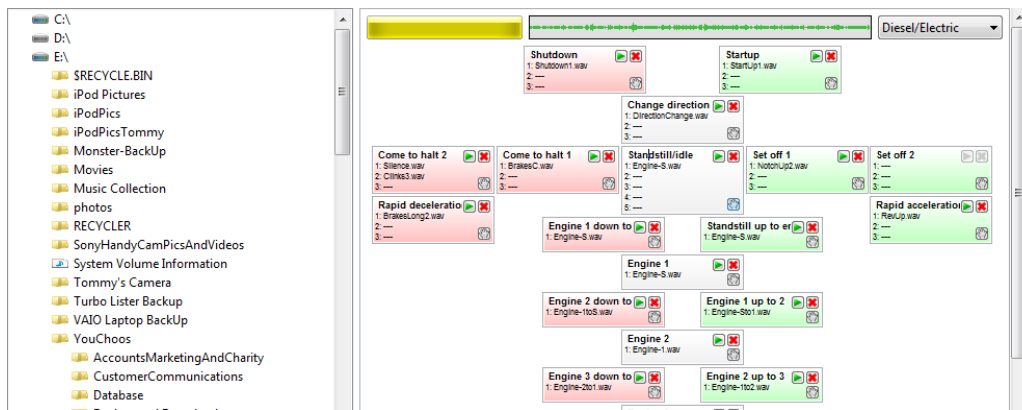
3 groups of chuffs are required – one group is used during acceleration, or under load, another group for deceleration, or download, and a third group for constant speed. It is common to make the acceleration chuffs louder than the constant speed chuffs, and to put rod clank sounds into the deceleration slots.

There are also individual slots available for defining the sound that plays *between chuffs*. This will normally be a background hiss, similar to the STANDSTILL/IDLE sound, and you will most likely point these slots all to the same sound. Be careful to make this sound a good looping sound, with no *clicking* or obvious *looping* effects – same principle as the main STANDSTILL/IDLE sound.

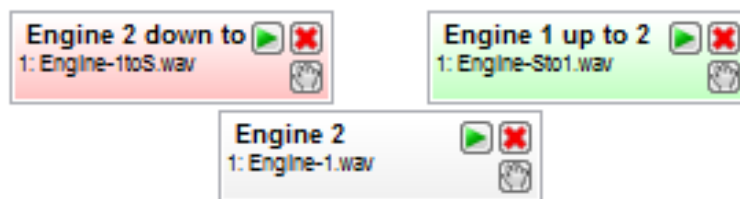
Diesel/Electric

Diesel/Electric engine sounds need to be defined rather differently, and the ability to graduate between different engine tones is possible throughout the speed range. You can choose how many *partitions* of engine tone you want to use, and in fact you can actually get away with just a single engine tone if you wish! This can actually be quite effective, since the decoder will automatically alter the pitch/frequency of the engine tone as it increases speed (defined by CVs 135 and 136 – described later in this guide).

However, a good sound Project will use different engine tones for different speeds. Often 2 or 3 partitions through the speed range is plenty, but you can define up to 9 in this application.

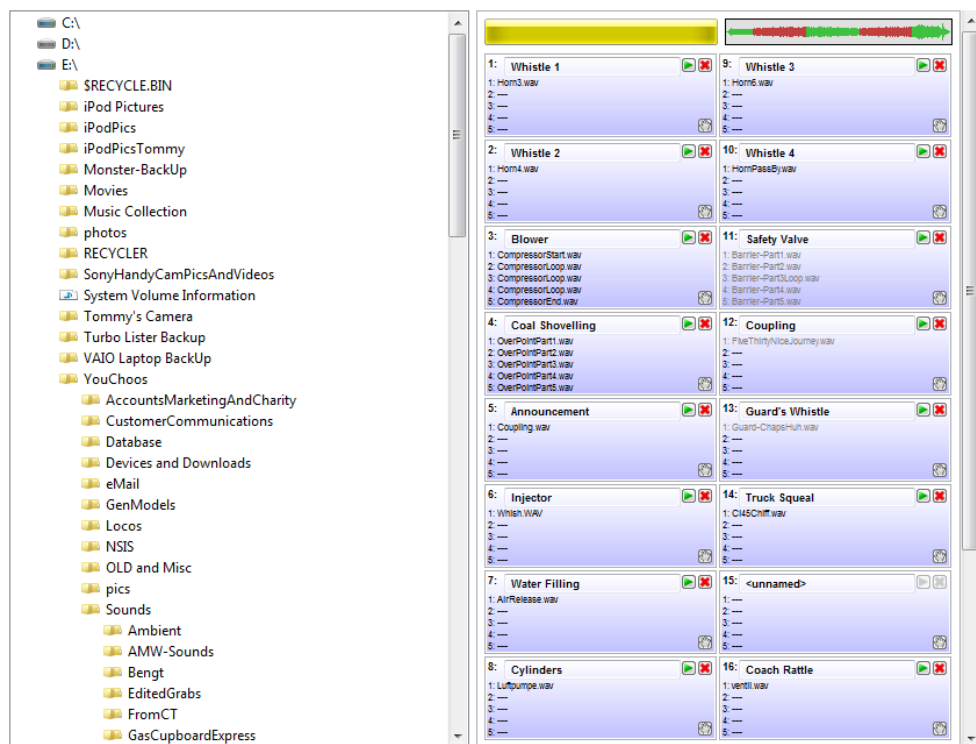


For each partition you will need 3 WAV files - one for the sound of the engine at that speed, then one for the transition between the lower partition and this one, and finally, a WAV for the transition from this sound back down to the lower speed. Many audio sound editors provide feature for blends and cross-fading from one sound to another, which you'll have to use to achieve this result:



Callable Sounds

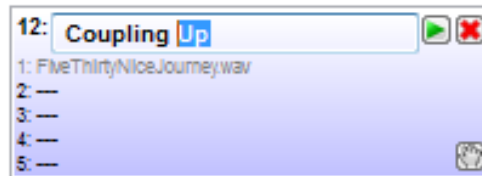
Callable sounds are those that are possible to invoke by either pressing a FunctionKey (see *Function Key Mapping* tab), or are playing through *random play* (see *Sound Config* tab).



16 sound effects are available for this purpose, although only sounds 1-8 are possible to configure for random play. Sounds 9-16 are accessed when the *ALT* feature is switched on – see the *Function Key Mapping* tab for more information on ALT.

All 16 sounds consist of up to 5 individual WAVs. If you fill all of them, then it is possible to make them *looping* sounds. Looping is configured on the *Sound Config* tab.

You can *name* each of these sound effects, for convenience, which helps when you move to the *Sound Config* tab. Simply click in the title and edit the field.





Sound Configuration

Callable Sound Effect 'Banks'

The *Callable* sound effects are grouped into 2 *banks* of sounds: 1-8 and 9-16. Sound 1 shares its configuration with Sound 9. Sound 2 shares its config with Sound 10 and so on. The list of these Callable sounds is presented in the *Sound Config* tab, giving various options:

- *Random Play* either at standstill, or during motion (you wouldn't want to hear the sound of water filling while the train is moving!!)
- *Volume*, ranging from 0 (silent) to 3 (full volume)
- Whether the sound is *Looping* – when the *middle* sound of the 5 slots is playing a number of times before the sound continues with the 4th and 5th slots. This applies to sounds that are played by a FunctionKey – as long as the Function is switched on, the sound continues to loop.
- *Middle sound # Reps* – defines how many times to repeat the middle sound when the sound is played through *Random Play*.
- *Overall # Reps* – defines the number of times the entire sound (all 5 slots) are repeated when activated through *Random Play*.

Callable Sound Effects									
Sound Effect	Effect Description	Random at Standstill	Random in Motion	Volume (0-3)	Looping	Middle sound # Reps (0-7)	Overall # Reps (0-7)		
Effect 1/9	Whistle 1/Whistle 3	<input type="checkbox"/>	<input type="checkbox"/>	3	<input type="checkbox"/>	0	1	<input type="checkbox"/>	<input type="checkbox"/>
Effect 2/10	Whistle 2/Whistle 4	<input type="checkbox"/>	<input type="checkbox"/>	3	<input type="checkbox"/>	0	1	<input type="checkbox"/>	<input type="checkbox"/>
Effect 3/11	Blower/Safety Valve	<input checked="" type="checkbox"/>	<input type="checkbox"/>	3	<input checked="" type="checkbox"/>	3	1	<input type="checkbox"/>	<input type="checkbox"/>
Effect 4/12	Coal Shovelling/Coupling	<input checked="" type="checkbox"/>	<input type="checkbox"/>	3	<input checked="" type="checkbox"/>	3	1	<input type="checkbox"/>	<input type="checkbox"/>
Effect 5/13	Announcement/Guard's Whistle	<input type="checkbox"/>	<input type="checkbox"/>	3	<input type="checkbox"/>	1	1	<input type="checkbox"/>	<input type="checkbox"/>
Effect 6/14	Injector/Truck Squeal	<input type="checkbox"/>	<input type="checkbox"/>	3	<input checked="" type="checkbox"/>	3	1	<input type="checkbox"/>	<input type="checkbox"/>
Effect 7/15	Water Filling/<unnamed>	<input type="checkbox"/>	<input type="checkbox"/>	3	<input type="checkbox"/>	0	1	<input type="checkbox"/>	<input type="checkbox"/>
Effect 8/16	Cylinders/Coach Rattle	<input type="checkbox"/>	<input type="checkbox"/>	3	<input type="checkbox"/>	0	1	<input type="checkbox"/>	<input type="checkbox"/>

Steam Sound Options

Various options are available for tweaking the way that steam engine sounds are played back. This includes the ability to synchronise chuffs with an external cam sensor (CV49 Bit0).

You may find it useful to experiment with *Weak Chuff Time* and *Strong Chuff Time*, as too low a value in these CVs can result in erratic-sounding chuffs, where the Back EMF effect causes the chuffs to wildly change from acceleration-to-deceleration-to-constant for no apparent reason!

The actual speed of the chuffs is automatically calculated relating to the current speed step (unless you are using an external cam sensor), and the rate of chuffs can be tweaked by experimenting with the *Time Between Chuffs* CVs. It is unlikely that you'll get these perfect, even with lots of

experimentation, but it is usually possible to get some approximately correct for the most common speed ranges that you tend to run the loco at.

Steam Sound Options		
<input type="checkbox"/> Sync chuffs with reed switch input		(CV49 Bit0)
<input type="checkbox"/> No chuffs on download/deceleration		(CV49 Bit4)
Weak chuff time	4	(CV130 - in 0.5sec units)
Strong chuff time	4	(CV146 - in 0.5sec units)
Time between chuffs at full speed	70	(CV132 in ms)
Time between chuffs at Speed Step 1	153	(9secs)

Other Options for Sound

There are numerous other options that affect various aspects of sound playback. Hopefully the descriptions are mostly self-explanatory. If not, please refer to the CV Reference for more information.

Generic Sound Options		
<input type="checkbox"/> No start-up sound		(CV49 Bit6)
<input type="checkbox"/> No final brake sound		(CV49 Bit7)
<input type="checkbox"/> Play start-up sound before moving		(CV137 Bit5)
Main running sounds volume	63	(CV121 0-63)
Rapid braking threshold (start)	12	(CV62 speed steps reduced in 1sec)
Rapid braking threshold (end)	25	(CV107 speed step *2)
Final brake threshold	20	(CV104 speed step *2 or 0=auto)
Rapid acceleration threshold	30	(CV177)
Load dependency	8	(CV110 0=none then 1=high->15=low)
Time between random sounds	50	(CV131 in 0.5sec units)
Pitch adjust - lowest	128	(CV135 where 128 is original pitch)
Pitch adjust - highest	160	(CV136 where 128 is original pitch)



Function Key Mapping

CT decoders use a special *Extended Function Mapping* procedure, which partially follow NMRA standard, but augment them to support the additional features of these sound decoders.

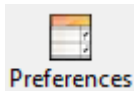
The *Function Key Mapping* tab present these features and a series of checkboxes that allow you to choose which FunctionKey on your digital DCC controller perform what feature in the decoder.

This includes all of the *Callable* sound effects, as well as the function outputs, plus the ability to switch the main running sounds on/off, the shunting mode on/off and the ALT feature on/off.

The ALT features is unique to CT decoders, and allows digital DCC controllers with a limited number of FunctionKey to access more of the *Callable* sound effects. By assigning a specific Fkey to the ALT feature, this switches between 2 banks of sounds (1-8 or 9-16). For example, in the screen shot below, F12 will normally sound "Whistle1", but with ALT activated (via F4 in the config), F12 will play "Whistle3" instead.

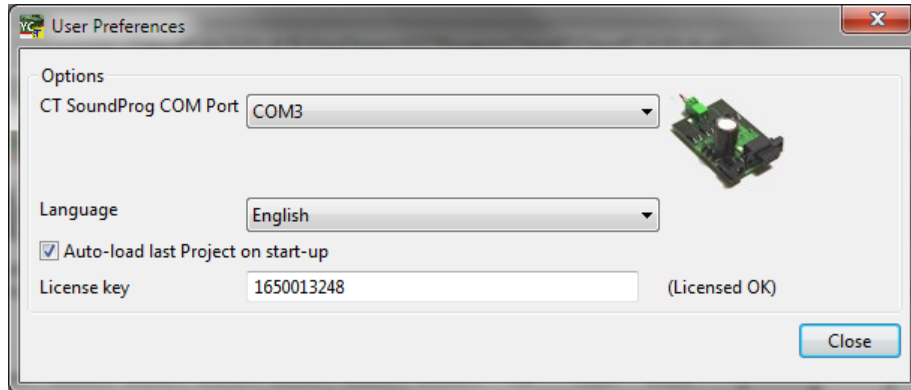
Effectively, ALT doubles the number of sounds that you can access, although you obviously have to remember whether the ALT feature is switched on or off at any time during operation to know which bank of sounds you are currently accessing!

	Whistle 1/Whistle e3	Whistle 2/Whistle e4	Blower/ Safety Valve	Coal Shovelling/ Coupling	Announ cement /Guard s	Injector /Truck Squeal	Water Filling/ <unna med>	Cylinder s/Coach Rattle											
FKey	Sound 8	Sound 7	Sound 6	Sound 5	Sound 4	Sound 3	Sound 2	Sound 1	Alt	Main	Shunt	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1	Output 0
F0 fwd				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
F0 rev				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F1				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F2				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F3				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F5	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F7	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F8	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F9	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F10	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F11	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
F12	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>



Preferences

Press the PREFERENCES toolbar button to access the Preferences dialog:



Here is where you can specify the COM port on which your CT SoundProg device is connected. This affects the .bat scripts that are created when you perform a LOAD operation or Firmware, or when you load sound Projects into a decoder.

You may also specify which *Language* to operate the application in. At the time of writing English and German are available. If you would like a translation in your own language then please contact YouChoos to discuss requirements – I am quite happy to supply you with a list of words/phrases, which you can translate, then I can include the translation into the product!

Finally, the Preferences dialog is where you enter your *License Key*. This is a numeric value as supplied to you by YouChoos, unique for your PC, and based on your PC's Computer Name.

Version History

V1.0 20th July 2012 – Initial Release

V1.1 29th July 2012 – Bug fix – crash when dragging WAV files into schedule of a blank project

V1.2 1st September 2012 – Bug fix – crash when dragging WAV files into schedule of a blank project (in additional to previous fix)

V1.3 1st October 2012 – Corrected calculation of size of project, so it is easily visible if the project will not fit on the decoder at load time. Also corrected support for 8-bit or 16-bit sounds in 22kHz or 11kHz formats.

V1.4 12th March 2013 – Added German translation, courtesy of Carsten & Christian at 1000-Digital.de.

V1.5 27th May 2013 – Changed default for CV49 to 10 for diesel projects to avoid 'drop-out' of the engine tone one stopped of the motor (thanks for Alfred Nusser for this hint).

Also changed pictures of decoders, SL75 and SL76 pictures courtesy of 1000-Digital. SL51-4 MTC, and SL82 4A courtesy of CT.

V1.6 3rd July 2013 – changed firmware suggestions for SL51-4 to use V127 firmware, which is required in order to successfully load sound projects on Win7 or 64-bit Windows reliably.

CV Reference

Below are listed all of the CVs available in CT decoders from Version 40 onwards. Note that some of them will not be relevant for all decoders. For example, the GE75 has not motor outputs, so none of the motor-related CVs will be relevant.

CV	Explanation	Range
1	Locomotive address: For short addresses when CV29 Bit 5 is set at 0.	1-127
2	Starting voltage: Voltage to motor at speed step 1. Tune this if your motor requires a little more <i>umph</i> to get it going, or you may find that the start-off sounds are not totally synchronised with the actual physical moving off of the loco.	0-255
3	Rate of acceleration – adjust to affect the <i>inertia</i> effect of speeding up. Set to 0 if you want no inertia effect – you control the speed immediately with your controller instead.	0-255
4	Rate of deceleration – similar to the above CV3, but for slowing down	0-255
5	Maximum speed: 0 for no artificial limit (maximum 255 has same effect as 0)	0-255
6	Middle speed: together with CV2 and CV5 a three-point speed curve can be set. Set CV6 = 0 to give a linear speed curve.	0-255
7	Version Number – read-only	variable
8	Manufacturer ID: CT Elektronik=117 – read-only	0-255
9	Motor PWM: 13-63 stepless from 30-150Hz, 141-191 16kHz for coreless and bell anchor motors	60-63 134-191
13	Analog mode: Use bits 0-3 to determine which function outputs (1-4) are switched on when operating on DC.	0-255
17+18	Extended address: CV29 bit 5 must be set in order to use a long address. CVs 17+18 are used to specify the long address. Please refer to NMRA standards for how this value is calculated.	128-10240
19	Multi-Unit (Consist) address	1-127
29	Miscellaneous configuration bits: Bit 0 (1) – Direction: OFF=normal; ON=inverted Bit 1 (2) – Speed steps: OFF=14/27; ON=28/128 Bit 2 (4) – Operating mode: OFF=digital only; ON=DC and Digital Bit 3 (8) – not used Bit 4 (16) – Speed curve: OFF=default speed curve using CVs 2, 5 & 6; ON=free speed curve using CVs 67-94. Bit 5 (32) – Address selection: OFF=1-127 (uses CV1); ON=128-10240 (uses CV17+18) Bit 6 (64) – not used Bit 7 (128) – not used	0-255
30	Error diagnosis: 1=motor; 2=light; 3=both short-circuit	0-3
33-42	Function mapping: according to NMRA for F0-F7, CV33-42=0. Please refer to CT extended function mapping document for more information.	0-255
43-46	Function mapping: according to NMRA for F8-F11 CV43-46=0. Please refer to CT extended function mapping document for more information.	0-255
49	Configuration bits for sound: CV49=0 gives 4 cylinder steam engine Bit 0 (1) – set if you use a Reed switch for wheel synchronising for steam engines - see also CV133=number of Reed Contacts – pulses per stroke e.g. CV133=1 means 1 stroke/pulse Bit 1 (2) – (OLD) set for Diesel or Electric loco (for use with the AUTO sound slots) Set both Bits 0 & 1 (value 3) for diesel and electric sounds that use Sound Slots 00 to 11 (must all be filled with sounds) – this is deprecated by the use of the AUTO sound slots (also known as Slots 1000-3000), so is not recommended. Bit 2 (4) gives 2 cylinder steam Bit 3 (8) gives 3 cylinder steam NOTE: for diesel and electric engines, an undocumented value of 10 for CV49 gives a better result, so CV49=10 is recommended for all diesel and electric projects. Bit 4 (16) – no steam strokes during downhill/deceleration (only idle sound) Bit 5 (32) – evaluate the LGB pulse from F1 Bit 6 (64) – play no sound between stand-still and running e.g. whistle (Slots 21-23 and Slots 24-26) Bit 7 (128) – no sound between running and stand-still e.g. brakes (Slots 27-29 and Slots 30-32)	0-255
50	EMF intensity: how strong is EMF effect: 0=no influence; 255=maximum. If you plan to use locomotives in a consist then use a lower value. This reduces the effect of locos working against each other if they cannot be configured to perform equally. Combine this value with CV51 and CV52 to cater for different motor types – often very smooth running can be achieved by experimenting with these values, even on motors that are apparently jerky to begin with under DCC.	0-255

51	P-Value: optimises EMF characteristics. Modify this to adapt to specific motor requirements (proportional part).	0-255
52	I-Value: optimises EMF characteristics. Modify this to adapt to specific motor requirements (integral part).	0-255
53	Special CV for programming beyond CV99 on old system PLUS decoder LOCK/UNLOCK feature For Roco Lokmaus users and any other systems which can only address CVs up to 99. Set bits 0 or 1 to address CVs over 99. Not necessary on most DCC systems. CV53=66: programming and feedback off (LOCK decoder from accidental re-programming...set to 77 to UNLOCK) CV53=77: programming and feedback on (UNLOCK decoder for programming... after setting to 66 previously) CV53=1: 100+ programmed CV value CV53=2: 200+ programmed CV value	0-255
54	PWM for function output: specifies the level of dimming applied to any function output with dimming selected via CV 57. Note that any function outputs with dimming switched on have the same dimming level applied – they are not individually dimmable by different amounts. Useful for reducing brightness of lights, or level of smoke generator for example. CV54=50 means 50% power output on function.	0-100
55	PWM for decoupler: represents the 'holding' current for the decoupler i.e. the reduced power for holding after the uncoupling impulse. Any function output defined as a decoupler (CV58) will initially get a higher-current pulse sent to that output when the function is activated, designed to 'kick' the magnet of a decoupler into motion, but then after a period (defined in CV56) it will be reduced to the level specified by this CV55 to hold it. Specified as a percentage.	0-100
56	Decoupler pulse time: how long is the impulse on the decoupler with full power until it is reduced to the value defined in CV55. Time is set in 0.1 second units.	0-255
57	Dimming mask: turns dimming (level defined in CV54) on and off for each function output. Each bit represents one function output, up to 8 functions (bit 0 for white wire, bit 1 for yellow wire etc.)	0-255
58	Dimming mask for decoupler function: defined which outputs should have decoupler processing applied. Each bit represents one function output. Any output specified here will have the effects of CV55 and 56 applied.	0-255
59	Signal controlled speed: "L" only available in ZIMO environment	0-255
60	Signal controlled speed: "U" only available in ZIMO environment	0-255
61	Signal controlled acceleration reaction time: only available in ZIMO systems	0-255
62	Braking threshold: indicates the number of speed steps that need to be reduced within 1 second in order for the <i>rapid braking</i> noise to trigger (Sound Slots 120 to 122). If the sounds are in three parts, the middle section is played in a loop until the delay is over, or the end threshold defined in CV107 is reached.	0-255
64	Reference voltage: EMF 100 = 20V track voltage	0-255
67-94	Free speed curve: activated with CV29 bit 4 is set. Default values: 9, 18, 27, 36, 45, 54, 63, 72, 81, 90, 99, 108, 117, 126, 135, 144, 153, 162, 171, 180, 189, 198, 207, 216, 225, 234, 243, 252	0-252
96	Zimo Signal controlled speed : "FL" speed selected between F-L (or MX9 HLU) is in version 52, see CV59, 60	0-255
97	Zimo Signal controlled speed : "LU" speed selected between L-U (or MX9 HLU) is in version 52, see CV59, 60	0-255
98	Zimo Signal controlled speed : "U-Stop" speed selected U-Stop (or MX9 HLU) is in version 52, see CV59, 60	0-255
104	End sequence braking threshold: determines at what point during deceleration the 'come to halt' sounds are triggered (slots 27-29 and slots 30-32). For example, if CV104 = 50 (= desired speed step *2) then the 'come to halt' sounds are triggered as the speed decreases from step 25 to 24. From SW Version 100, you can set CV104 = 0, which causes this sequence to be calculated automatically depending on the length of the recorded sound. See also CV62 and CV107.	0-255
105	User CV: free for remembering purchase date or similar user information	0-255
106	User CV: free for remembering purchase date or similar user information	0-255
107	Rapid braking end threshold: if a rapid braking sound is defined as looping in Slots 120-122 (also see CV62 for when rapid braking sound is triggered), it will continue to loop until the desired speed step is reached, or when the speed step defined here in CV107 is reached. This gives the possibility to stop the rapid braking sound below a certain speed. CV107=50 (= desired speed step *2) stops the rapid braking sound below speed step 25. Applies to SW Version 40 onwards.	0-255
108	Bitmask for endless manual sound: ONLY operational when CV49 bit 5 is set. For use with LGB pulse chains. Bit 0 for sound 1; bit 1 for sound 2 etc.	0-255
109	Selection of CV set: bit 0=0 gives CV set 1; bit 0=1 gives CV set 2. Can be used for various purposes – CV set for home layout + CV set for club layout for example. Hard reset will only affect the currently selected CV set. CV109 will be unchanged by a hard reset.	0-1
110	Load-dependent sound variation: set to 0 to give no load-dependent variation; then a range from 1 (high dependency) to 15 (low dependency on load). If configured carefully, this feature can be used to detect a heavy train, or uphill climbs and cause the sound (chuffs for steam) to use the 'acceleration' sounds under that load. However, beware – using an excessive value here will cause the train to produce unnecessarily loud chuffs at unwanted time e.g. travelling around a curve.	1-15
111	Intensity of acknowledgement pulse (ACK): improves the programming capability, 128 is approx 50% of max acknowledgement pulse (motor dependent) 200 = normal.	0-255

112	Random sounds at standstill: chooses which sounds may occur randomly while idling. Set to 0 for no random sounds at standstill or set bits 0-7 to control which sounds are included in random play (Effects 1-8). See also CV131. It is not possible to specify sounds 9-16 for random play.	0-255
113	Random sounds during motion: chooses which sounds may occur randomly while the loco is moving. Set to 0 for no random sounds while moving or set bits 0-7 to control which sounds are included in random play (Effects 1-8). See also CV131. It is not possible to specify sounds 9-16 for random play.	0-255
114	PWM for effects – some of the lighting effects pulse between a high point and a low point. This CV specifies the level of the low point in those cycles. See also CV154 to 161.	0-100
115	Cycle time for flashing light effects, as specified by CV154-161	0-255
116	Shunting function (yard mode) configuration (shunting mode may be switched on/off via appropriate function mapping – see CT extended function mapping document for more information): Only active if bit0-bit2 set Bit 0 (1) – effects of CV3 and CV4 are disabled when shunting mode activated Bit 1 (2) - maximum speed is halved Bit 2 (4) – reverse is 65% of maximum speed (regardless of shunting mode) - applies from SW Version 40 or later Bit 3 (8) - brakes with diode 4:1 is active – see also CV 162 Bit 4 (16) - brakes without diode Bit 5 (32) - not used, must always be 0 Bit 6 (64) - means that the “Rangierfunktion” acts as a command button that is, that the automatic train control system (brake or diode and HLU) - (equivalent to the MAN key = manual) Bit 7 (128) – short burst of idling, returning to normal speed	0-255
117	Defines the Function key that causes the <i>Low-Beam</i> dimming effect to be applied. 1=F1, 2=F2 ... 12=F12. See also CV118, 119 and 120.	1-12
118	Mask to specify which function outputs have <i>Low-Beam</i> dimming applied when activated by the function key specified in CV 117. Each bit represents 1 function output (0-7). 1 = white wire, 2 = yellow wire, 4 = green wire, 8 = purple wire, 16 = 5 th output, 32 = 6 th output, 64 = 7 th output, 128 = 8 th output	0-255
119	PWM dimming level for <i>Low-Beam</i> dimming as defined by CV117 and CV118. A value of 50 = approximately 50% of full brightness, 100 = 100% i.e. no dimming. 0 also means no dimming.	0-100
120	Cycle duration of how long the <i>Low-Beam</i> dimming will last – see CVs 117, 118 and 119.	0-255
121	Volume for main sound on. Values depend upon software version: Old versions have range 0-3 only (0 gives no sound, 3 gives maximum volume). Newer versions have range 0-63 (63 being maximum). Expect that future versions may range from 0-255.	0-63
122	Volume & Repetitions for sound effect 1 (Slots 37-41): bits 0-1 specify volume; bits 2-4 for number of repetitions of middle sound (Slot 39); bits 5-7 define number of repetitions of sound effect overall. If the ‘Alternate’ sounds are activated, this applies to sound effect 9 instead (Slots 77-81)	0-255
123	Volume & Repetitions for sound effect 2 (Slots 42-46): bits 0-1 specify volume; bits 2-4 for number of repetitions of middle sound (Slot 44); bits 5-7 define number of repetitions of sound effect overall. If the ‘Alternate’ sounds are activated, this applies to sound effect 10 instead (Slots 82-86)	0-255
124	Volume & Repetitions for sound effect 3 (Slots 47-51): bits 0-1 specify volume; bits 2-4 for number of repetitions of middle sound (Slot 48); bits 5-7 define number of repetitions of sound effect overall. If the ‘Alternate’ sounds are activated, this applies to sound effect 11 instead (Slots 87-91)	0-255
125	Volume & Repetitions for sound effect 4 (Slots 52-56): bits 0-1 specify volume; bits 2-4 for number of repetitions of middle sound (Slot 54); bits 5-7 define number of repetitions of sound effect overall. If the ‘Alternate’ sounds are activated, this applies to sound effect 12 instead (Slots 92-96)	0-255
126	Volume & Repetitions for sound effect 5 (Slots 57-61): bits 0-1 specify volume; bits 2-4 for number of repetitions of middle sound (Slot 59); bits 5-7 define number of repetitions of sound effect overall. If the ‘Alternate’ sounds are activated, this applies to sound effect 13 instead (Slots 97-101)	0-255
127	Volume & Repetitions for sound effect 6 (Slots 62-66): bits 0-1 specify volume; bits 2-4 for number of repetitions of middle sound (Slot 64); bits 5-7 define number of repetitions of sound effect overall. If the ‘Alternate’ sounds are activated, this applies to sound effect 14 instead (Slots 102-106)	0-255
128	Volume & Repetitions for sound effect 7 (Slots 67-71): bits 0-1 specify volume; bits 2-4 for number of repetitions of middle sound (Slot 69); bits 5-7 define number of repetitions of sound effect overall. If the ‘Alternate’ sounds are activated, this applies to sound effect 15 instead (Slots 107-111)	0-255
129	From SW version 40 onwards CV 129 has this meaning: Volume & Repetitions for sound effect 8 (Slots 72-76): bits 0-1 specify volume; bits 2-4 for number of repetitions of middle sound (Slot 74); bits 5-7 define number of repetitions of sound effect overall. If the ‘Alternate’ sounds are activated, this applies to sound effect 16 instead (Slots 112-116) Note that it is not possible to individually control the volume or looping of sound effects 9-16 (Slots 77-116) – they take on the settings for the equivalent effect in sounds 1-8.	0-255

	<p>Prior to SW version 40, CV 129 had the following meaning: Strong time when the sound after acceleration is being strongly reduced (0.5 second units), valid for sounds in Slots 0-3 (this has moved to CV 146 from SW Version 40 onwards)</p>	
130	Weak time – how long the deceleration chuffs (Slots 8-11) should continue for after reducing speed (0.5 second units).	0-255
131	Random time: minimum time between 2 random sounds (0.5 second units)	0-255
132	Stroke speed at full speed: Time between 2 strokes/chuffs at full speed	0-255
133	<p>Stroke speed at Step 1 (LOW byte): time between two steam strokes at speed step 1 in seconds. This can be a large number, so is defined using 2 CVs (133 and 134 with the LOW byte in CV133 and HIGH byte in CV134). Calculated via the formula: $K = 1476 / \text{time}$ (1476 is a constant always used for this calculation by the decoder) Example 1: $K=153$ gives approximately 9.6 sec between chuffs Example 2: if 20 seconds wanted between chuffs, so $K = 1476 / 20 = 73.8$ rounded to 74 gives CV133 = 74, CV134 = 0 Example 3: if 3 seconds wanted between chuffs, so $K = 1476 / 3 = 492$. Since $K > 256$ we need to use 2 bytes, so split low byte and high byte: $492 / 256 = 1.927875$ thus CV134 = 1 (rounded down), CV133 = remainder i.e. $492 - (256 * 1) = 236$</p>	0-255
134	Stroke speed at Step 1 (HIGH byte): time between two steam strokes at speed step 1 in seconds – see also CV133.	0-255
135	<p>Frequency min: reduces the pitch of chuffs or engine sounds at lower speed steps. 128 is default pitch (sounds played as originally recorded). Use this CV along with CV136 to define how engine pitch or chuff pitch increases with the speed of the loco.</p>	0-255
136	Frequency max: increases the pitch of chuffs or engine sounds at higher speed steps. 128 is default pitch (sounds played as originally recorded).	0-255
137	<p>Special CV Bit 0 (1) - OFF = 8 functions, ON = 14 functions (MAN-bit) refers to F0-F12, btw CV33-CV46 free assignment. Bit 1 (2) - ZIMO train number impulse on / off Bit 2 (4) - strong / normal / weak switched with F1, effective only when CV110 is active, and dimmable via CV54 Bit 3 (8) - strong / normal / weak switched with F2, effective only when CV110 is active, and dimmable via CV54 Bit 4 (16) - ZIMO speed contrl - dependent train control 0 = off 1 = on Bit 5 (32) - Start sequence (Sound Slots 21-23) is played before motor starts to spin – useful for diesel engines in particular as a rev-up sound before physically setting off Bit 6 (64) - Set ON if you want to control additional functions using F4 – press twice for F5, 3x for F6, 4x for F7 (useful if your controller does not have access to many functions directly) Bit 7 (128) - 32kHz frequency motor control from software version 41, factory Bit7 = 0 16kHz</p>	0-255
138	Break time (HLU): break delay for HLU section (for ZIMO systems only)	0-255
139	Short-circuit threshold 1: direct cut-off at overload of function outputs	0-255
140	Short-circuit threshold 2: fast cut-off at overload of function outputs	0-255
141	Short-circuit threshold 3: slow cut-off at overload of function outputs	0-255
142	Short-circuit threshold 1: direct cut-off at overload of motor output	0-255
143	Short-circuit threshold 2: fast cut-off at overload of motor output	0-255
144	Short-circuit threshold 3: slow cut-off at overload of motor output	0-255
145	<p>Activation of sound looping - if the corresponding sound function is activated, it will first play its 1st and 2nd Slots. After that, Slot 3 is repeated until the feature is turned off. After requesting off, it plays closing 4th and 5th Slots. Bit 0 (1) - for looping of Sound 1 (Slots 37-41) Bit 1 (2) - for looping of Sound 2 (Slots 42-46) Bit 2 (4) - for looping of Sound 3 (Slots 47-51) Bit 3 (8) - for looping of Sound 4 (Slots 52-56) Bit 4 (16) - for looping of Sound 5 (Slots 57-61) Bit 5 (32) - for looping of Sound 6 (Slots 62-66) Bit 6 (64) - for looping of Sound 7 (Slots 67-71) Looping for Sound effects 8-16 cannot be configured – they are not looping.</p>	0-255
146	<p>From SW Version 40 onwards, CV146 has the following meaning (used to be in CV129): Strong time: time that the acceleration chuffs sounds (Slots 0-3) continue after strong acceleration (0.5 seconds units). OLD: SW Version 39 and earlier CV146 had this meaning: CV146 = 1 means: Z3 (switching function) is connected with F7, 0 means inactive Z3 CV146 = 2 means: Z4 (switching function) is switched with F8, 0 means inactive Z4 CV146 = 4 means: Sound3 is connected with F7, 0 means inactive Sound3</p>	0-255

	CV146 = 8 means: Sound4 is connected with F8, 0 means inactive Sound4 CV146 = 16 means: Z5 (switching function) is switched with F9, 0 means inactive Z5 CV146 = 32 means: Z6 (switching function) is switched with F10, 0 means inactive Z6 CV146 = 64 means: Sound5 is connected with F9, 0 means inactive Sound5 CV146 = 128 means: Sound6 is connected with F10, 0 means inactive Sound6	
147	Discharge of the coupling: a kickback effect causing the locomotive to run backwards slightly – a tiny jerk of the motor to achieve uncoupling.	0-126
148	Away from wagons: speed when driving away from wagons, locomotive runs in the current direction, 126 = max. Speed under. Take into account the time set in CV3.	0-126
149	Discharge time: the time for the unit pushed back. 0.1 seconds, 10 = 1 seconds	0-255
150	Drive away: the time for driving away unit 0.1 seconds, 30 = 3 seconds	0-255
151	Selection of automatic disconnection: 0 = off, 1 = F1 2 = F2 3 = F3, 4 = F4, etc.	1-12
152	Uncoupling mask forwards: Select the function to be used, 4 = F2, 8 = F3, 16 = F4, 32 = F5, 64 = F6 128 = F7	0-255
153	Uncoupling mask backwards: Select the function to be used, 4 = F2, 8 = F3, 16 = F4, 32 = F5, 64 = F6 128 = F7	0-255
154	<p>Lighting effect for front light (output 0 white wire):</p> <ul style="list-style-type: none"> 0 - No effect 1 - Flashing 2 - Flash-pull 3 - Single pulse strobe 4 - Double Flashing strobe 5 - Headlight (brightness between maximum and PWM value in CV 114) 6 - Ditch light left (brightness between maximum and PWM value in CV 114) 7 - Ditch light right (brightness between maximum and PWM value in the CV 114) 8 - Rotary beacon (brightness between maximum and PWM value in the CV 114) 9 - Gyalite (brightness between maximum and PWM value in the CV 114) – can also be used for firebox glow 10 - Mars light 11 - Soft-start 12 – Brake sparks (short flash activated only when the loco comes to a stop) <p>For output when forward only, add 64 to the above value e.g. 1 + 64 = 65 flashing on Forward only. For output when in reverse only, add 128 to the above value e.g. 1 + 128 = 129 flashing on Reverse only. If neither 64 nor 128 is added, it is assumed that the effect should be active for both forwards and reverse. CVs 155 -161 provide the same effects for each of the other function outputs.</p>	0-255
155	Lighting effect for function output 1 (yellow wire) - see CV154	0-255
156	Lighting effect for function output 2 (green wire) - see CV154	0-255
157	Lighting effect for function output 3 (purple wire) - see CV154	0-255
158	Lighting effect for function output 4 (solder pad) - see CV154	0-255
159	Lighting effect for function output 5 (solder pad) - see CV154	0-255
160	Lighting effect for function output 6 (solder pad) - see CV154	0-255
161	Lighting effect for function output 7 (solder pad) - see CV154	0-255
162	Sensitivity of the diode voltage: see also CV116. Value of 10-20 is generally well tolerated, the smaller the value the more sensitive.	0-255
163-176	Extended function mapping: Please refer to CT extended function mapping document for more information.	0-255
177	Trigger for rapid acceleration: indicates the number of speed levels that must be attained within any 100 msec period in order to trigger the sound in Slots 123-125). If the sounds are in three parts, the middle part is played in a loop until the requested speed is reached. See also CV 107 for trigger to rapid braking sound.	0-252

CV.dat File Format

The CV.dat file is a simple text file with value pairs of CVNumber=Value. Values are in the range 0-255. CVNumbers must be sequential, and all CVs from 1-177 should be defined:

```
1=3  
2=3  
3=50  
4=50  
5=0  
6=0  
7=40  
8=117  
9=141  
10=0  
11=0  
12=0  
13=0  
...etc.
```


Sounds.txt File Format

Decoders from Version 40 onwards (this application does not support earlier versions) use a similar format for the Sounds.txt file. Comments are any line starting with a semi-colon character. The header of the file is always the same consisting of either 2 lines or 3 lines (The ***Mega168 line is only required for SL76/GE76 decoders).

Below is an example for a 4-cylinder steam loco. Note that any blank slots should be filled with "---" characters:

```
**16MBit
**4.Generation
**Mega168 (required for SL76/GE76 only, must not be included for others)
;STEAM: Chuffs during acceleration
00: Chuff-Acc-1-Mix.wav
01: Chuff-Acc-2-Mix.wav
02: Chuff-Acc-3-Mix.wav
03: Chuff-Acc-4-Mix.wav
;STEAM: Chuffs during constant speed
04: Chuff-Const-1-Mix.wav
05: Chuff-Const-2-Mix.wav
06: Chuff-Const-3-Mix.wav
07: Chuff-Const-4-Mix.wav
;STEAM: Chuffs/Clank during deceleration
08: Clank-1.wav
09: Clank-2.wav
10: Clank-3.wav
11: Clank-4.wav
;STEAM: Between acceleration chuffs
12: StandStill-Quiet.wav
;STEAM: Between constant chuffs
13: StandStill-Quiet.wav
;STEAM: Between deceleration chuffs/clanks
14: StandStill-Quiet.wav
;Startup
15: StartUp.wav
16: ---
17: ---
;Shutdown
18: ShutDown.wav
19: ---
20: ---
;Set off (channel 1)
21: Hiss1.wav
22: ---
23: ---
;Set off (channel 2)
24: ---
25: ---
26: ---
;Come to halt (channel 1)
27: 7F-Brakes-Halt.wav
28: ---
29: ---
;Come to halt (channel 2)
30: ---
31: ---
32: ---
;Standstill/idle
33: StandStill-Quiet.wav
```

34: ---
35: ---
36: ---
;Effect 1 - Whistle 1
37: Whistle-3.wav
38: ---
39: ---
40: ---
41: ---
;Effect 2 - Whistle 2
42: Whistle-1.wav
43: ---
44: ---
45: ---
46: ---
;Effect 3 - Blower
47: Blower-Start.wav
48: Blower-Middle.wav
49: Blower-Middle.wav
50: Blower-Middle.wav
51: Blower-End.wav
;Effect 4 - Coal Shovelling
52: coal1.wav
53: coal2.wav
54: coal3.wav
55: coal4.wav
56: coal5.wav
;Effect 5 - Announcement
57: BathGreenPark.wav
58: ---
59: ---
60: ---
61: ---
;Effect 6 - Injector
62: Injector2-Loop-P1.wav
63: Injector2-Loop-P2and4.wav
64: Injector2-Loop-P3.wav
65: Injector2-Loop-P2and4.wav
66: Injector2-Loop-P5.wav
;Effect 7 - Water Filling
67: water1.wav
68: water2.wav
69: water2.wav
70: water2.wav
71: water3.wav
;Effect 8 - Cylinders
72: Cy lind1.wav
73: Cy lind2.wav
74: Cy lind2.wav
75: Cy lind2.wav
76: Cy lind3.wav
;Effect 9 - Whistle 3
77: Whistle-2.wav
78: ---
79: ---
80: ---
81: ---
;Effect 10 - Whistle 4
82: Wl4.wav
83: ---
84: ---
85: ---
86: ---
;Effect 11 - Safety Valve
87: SafetyValve-3F-Loop-P1.wav

```

88: SafetyValve-3F-Loop-P2.wav
89: SafetyValve-3F-Loop-P3.wav
90: SafetyValve-3F-Loop-P4.wav
91: SafetyValve-3F-Loop-P5.wav
;Effect 12 - Coupling
92: Coupling.wav
93: ---
94: ---
95: ---
96: ---
;Effect 13 - Guard's Whistle
97: GuardWhi.wav
98: ---
99: ---
100: ---
101: ---
;Effect 14 - Truck Squeal
102: TrucksPart1.wav
103: TrucksPart2.wav
104: TrucksPart3.wav
105: TrucksPart4.wav
106: TrucksPart5.wav
;Effect 15 - <unnamed>
107: ---
108: ---
109: ---
110: ---
111: ---
;Effect 16 - Coach Rattle
112: CoachP1.wav
113: CoachP2.wav
114: CoachP2.wav
115: CoachP2.wav
116: CoachP3.wav
;Change direction
117: LittleHiss.wav
118: ---
119: ---
;Rapid braking
120: ---
121: ---
122: ---
;Rapid acceleration
123: RapidAcc1.wav
124: ---
125: ---

```

Diesel sound Projects are a little different – you must leave the Chuff slots blank (slots 00-14), but then define the AUTO slots at the end of the file. Each AUTO slot represents an engine *partition*. The more partitions (AUTO slots) you have, then more variation the engine will have through the speed range.

An AUTO slot comprises 3 WAV files where the 1st one is the transition from the next engine sound down, *up* to this one. The 2nd WAV is the constant sound for the engine at this partition. The 3rd WAV is the transition from this partition back *down* to the lower speed. You can supply a large number of AUTO slots if you wish, but using between 2 and 6 gives a good overall variation in engine tone through the speed range.

```

**16MBit
**4.Generation
**Mega168 (required for SL76/GE76 only, must not be included for others)

```

```

;STEAM: Chuffs during acceleration
00: ---
01: ---
02: ---
03: ---
;STEAM: Chuffs during constant speed
04: ---
05: ---
06: ---
07: ---
;STEAM: Chuffs/Clank during deceleration
08: ---
09: ---
10: ---
11: ---
;STEAM: Between acceleration chuffs
12: ---
;STEAM: Between constant chuffs
13: ---
;STEAM: Between deceleration chuffs/clanks
14: ---
;Startup
15: StartUp1.wav
16: ---
17: ---
;Shutdown
18: Shutdown1.wav
19: ---
20: ---
;Set off (channel 1)
21: NotchUp2.wav
22: ---
23: ---
;Set off (channel 2)
24: ---
25: ---
26: ---
;Come to halt (channel 1)
27: BrakesC.wav
28: ---
29: ---
;Come to halt (channel 2)
30: Silence.wav
31: Clinks3.wav
32: ---
;Standstill/idle
33: Engine-S.wav
34: ---
35: ---
36: ---
;Effect 1 - Whistle 1
37: Horn3.wav
38: ---
39: ---
40: ---
41: ---
;Effect 2 - Whistle 2
42: Horn4.wav
43: ---
44: ---
45: ---
46: ---
;Effect 3 - Blower
47: CompressorStart.wav
48: CompressorLoop.wav

```

49: CompressorLoop.wav
50: CompressorLoop.wav
51: CompressorEnd.wav
;Effect 4 - Coal Shovelling
52: OverPointPart1.wav
53: OverPointPart2.wav
54: OverPointPart3.wav
55: OverPointPart4.wav
56: OverPointPart5.wav
;Effect 5 - Announcement
57: Coupling.wav
58: ---
59: ---
60: ---
61: ---
;Effect 6 - Injector
62: Whish.WAV
63: ---
64: ---
65: ---
66: ---
;Effect 7 - Water Filling
67: AirRelease.wav
68: ---
69: ---
70: ---
71: ---
;Effect 8 - Cylinders
72: Luftpumpe.wav
73: ---
74: ---
75: ---
76: ---
;Effect 9 - Whistle 3
77: Horn6.wav
78: ---
79: ---
80: ---
81: ---
;Effect 10 - Whistle 4
82: HornPassBy.wav
83: ---
84: ---
85: ---
86: ---
;Effect 11 - Safety Valve
87: Barrier-Part1.wav
88: Barrier-Part2.wav
89: Barrier-Part3Loop.wav
90: Barrier-Part4.wav
91: Barrier-Part5.wav
;Effect 12 - Coupling
92: FiveThirtyNiceJourney.wav
93: ---
94: ---
95: ---
96: ---
;Effect 13 - Guard's Whistle
97: Guard-ChapsHuh.wav
98: ---
99: ---
100: ---
101: ---
;Effect 14 - Truck Squeal
102: Cl45Chiff.wav

103: ---
104: ---
105: ---
106: ---
;Effect 15 - <unnamed>
107: ---
108: ---
109: ---
110: ---
111: ---
;Effect 16 - Coach Rattle
112: ventil.wav
113: ---
114: ---
115: ---
116: ---
;Change direction
117: DirectionChange.wav
118: ---
119: ---
;Rapid braking
120: BrakesLong2.wav
121: ---
122: ---
;Rapid acceleration
123: RevUp.wav
124: ---
125: ---
AUTO
Engine-S.wav
Engine-S.wav
Engine-S.wav
AUTO
Engine-Sto1.wav
Engine-1.wav
Engine-1toS.wav
AUTO
Engine-1to2.wav
Engine-2.wav
Engine-2to1.wav

Acknowledgements

Information on how CT decoders operate has been gathered over a number of years from various sources, as well as through a lot of experimentation. Below are listed the people and sources of information that I have used:

- CT
- JMRI/DecoderPro
- NMRA – standards documents
- Nigel Cliffe (2mm Association)
- Arnold Huebsch / AMW
- 1001-digital
- Alfred Nusser